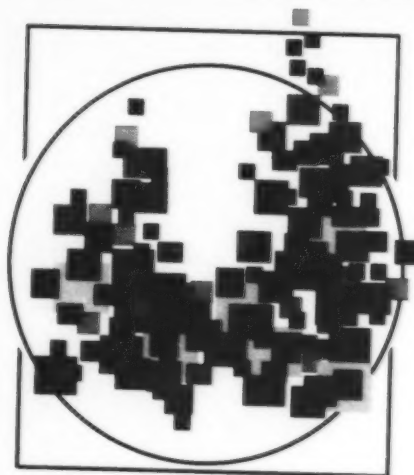




CENTRE FOR
SYSTEMS
SCIENCE

**An Intension-Based Model
Of Nested Epistemic
Processes**

Robert F. Hadley

CSS-IS TR 89-04

Revised on January 1, 1990

ABSTRACT

A process-oriented model of belief and knowledge is presented which permits the representation of nested propositional attitudes within first-order logic. The model (NIM, for 'nested intensional model') is axiomatized, intensionally based, and sanctions inferences involving nested epistemic attitudes, with different agents and different times. Through the expedient of context-sensitive semantic conventions, NIM attains broad expressive power without resort to encumbering quotation marks, concatenation functions, or "sense notation". Because NIM is grounded upon intensions, it provides a framework in which agents may reason about the beliefs of another agent while remaining neutral with respect to the syntactic forms used to express the latter agent's beliefs. In addition, the axioms of NIM may be used to *justify* conclusions which an agent derives concerning the beliefs of another agent. Because NIM makes explicit many of the process-related preconditions of belief acquisition, NIM underscores the limitations of inter-agent reasoning. Moreover, NIM provides agents with a conceptual map, interrelating the concepts of 'knowledge', 'belief', 'truth', and a number of cognate concepts, such as 'infers', 'retracts', and 'questions'. From a logician's standpoint, the language of NIM provides a useful tool for intensional logic. In addition, the inclusion of a distinguished truth predicate, and rules of inference for reasoning about *truth*, provide the language with an expressive power approaching that of natural language. The soundness and completeness of NIM as a formal, first-order theory are established. Finally, it is shown that, by varying an *intensional* parameter, NIM may be used either as a human-like *cognitive* model, or as a model of the epistemic processes of purely rational artificial agents (including "truth maintenance systems").

1. INTRODUCTION

Within the cognitively related disciplines (including AI) there are three dominant approaches towards the modeling of epistemic states.¹ For convenience we call these the logistic, syntactic, and intensional approaches. The *logistic* approach seeks to discover a *logic* of belief (knowledge) which would enable one to derive, *a-priori*, the totality of things an agent believes (or knows), given the formulae explicitly written in the agent's belief base.² The *syntactic* approach (as Levesque, 1984, dubs it) identifies an agent's beliefs with just those formulae explicitly written in the agent's belief base -- the agent has no other beliefs. By contrast, the *intensional* approach identifies beliefs with abstractions, e.g., with *thought contents* or propositions. On this account the same belief may be expressed by means of a limited variety of distinct syntactic objects. As will emerge in what follows, our present concerns lie with the intensional model, and with how intensions are related to the *processes* of inference, belief acquisition, and belief retraction. In section 2 we present a *nested intensional model* of belief (NIM) which sanctions inferences among nested epistemic propositions, while being sensitive to the empirical contingencies of epistemic processes. Section 3 illustrates the application of NIM to *inter-agent* epistemic reasoning. In sections 4 and 5 we describe the relationship of NIM to augmented truth maintenance systems (ATMS) (de Kleer 1986; Brown, Gaucas, and Benanav 1987), and demonstrate that NIM may be regarded as a *formal model* of the state transitions of an ATMS belief base. Unlike earlier process theories (cf. Goodwin, 1985) NIM may (by varying an *intensional* parameter) be used either as a human-like *cognitive* model, which addresses the fact that the same belief may be expressed by distinct formulae, or as a model for the epistemic processes of purely rational artificial agents. However, before examining the details of NIM, we briefly examine the logistic and syntactic approaches to belief, and consider how these differ from the present work.

1.1 The Logistic Approach

Those who adopt the logistic approach are concerned with the discovery of *inference rules*, such that an agent's explicit beliefs are *closed under* the application of these rules. Typically, the epistemic logics proposed do not require an agent's beliefs to be closed under *all* valid inference rules (full logical omniscience is not required), but closure under *certain* inference rules is still required. Thus, these logics typically require that agents each possess an infinity of beliefs, or at least believe an infinity of distinct formulae. (cf. Levesque, 1984; Halpern & Moses, 1985; Lakemeyer, 1987). Some have questioned the

¹A seven page precursor of this paper was presented to the Tenth Annual Conference of the Cognitive Science Society in Montreal, 1988.

²Hereafter, "belief" shall be taken to include knowledge, unless the context requires that the distinction be explicit.

motivation for introducing logics which represent agents as believing an infinity of formulae (e.g., Fagin & Halpern, 1985), but the underlying rationale may be that such logics employ a non-standard sense of belief, which yet has application in the context of finite agents. For example, the sense of belief appropriate to (Levesque, 1984) *might* be such that " p is believed" means approximately that one could *eventually come to believe* p by a tractable computational process.³ However, whether or not one thinks that epistemic logics (of the "infinite belief set" variety) require us to adopt a non-standard sense of belief, such logics usually are not concerned with certain important *process-related* aspects of belief. For example, with limited exceptions⁴ such logics do not address the fact that an agent acquires and *changes* beliefs over time, or that these processes often involve inference, testing beliefs for consistency, retraction, etc. Doubtless, most epistemic logics were never intended to address those aspects of belief, but for certain applications one desires a *formal model* of how these cognitive processes relate to the belief states of the agent. An agent X who knows what another agent Y believed an hour ago may wish to form a reasonable conjecture about what Y believes *now*. If so, X needs to know what *processes* must be assumed to exist in order to arrive *even* at a reasonable conjecture about Y 's present beliefs. Of course, some aspects of belief change are addressed by truth maintenance systems (e.g., de Kleer, 1986; Doyle, 1979), but TMSs are concerned with *procedures* for manipulating belief sets, rather than with formal models which interrelate the concepts of belief, knowledge, inference, retraction, etc. By contrast, we are here concerned with formalizations of epistemic states, and the processes which affect those states.

1.2 The Syntactic Approach

Now, apart from epistemic *logics* which posit an infinity of beliefs, certain *syntactically-oriented* belief models do address the fact that beliefs can be acquired and retracted by *processes*. Both Goodwin (1985) and Doyle (1983) present process theories for those non-monotonic reasoning systems which involve the use of a TMS. However, in both these theories the assumption is made that any agent who infers a given conclusion necessarily comes to believe that conclusion (at least temporarily). This assumption appears to preclude the possibility that a sophisticated agent may choose to investigate whether the conclusion conflicts with other long-standing beliefs, before "committing" to that conclusion. Moreover, the process theories of Goodwin and Doyle share a crucial aspect with other syntactic approaches to belief (e.g. Hass 1985, 1986; Maida, 1986), namely, they identify beliefs with particular syntactic objects, so that, for example, an agent might believe the formula $p \vee q$ but not $q \vee p$, or believe "Tadpoles swim" but not

³I am indebted to Jim Delgrandge for suggesting this sense of 'belief'.

⁴Konolige, (1985) presents an epistemic logic which does address the process of belief *introspection*. However, Konolige's system is not intended to address most other process-related aspects of belief.

"Pollywogs swim". Now, many reject the syntactic approach entirely, on the grounds that it is too fine-grained. The syntactic approach arguably fails to account for the fact that *some* syntactically distinct formulae seem to express precisely the same thought, and therefore ought to have the same belief status (cf. Levesque, 1984; Hadley, 1986). No doubt, there are particular applications where one can safely ignore the fact that syntactically distinct formulae sometimes express the same belief, but for other applications this is unwise. As we shall see, in the context of an ATMS, it is desirable to identify *different* formulae which express the same belief with a single node in the belief base. Doing so facilitates the goal of maintaining the *consistency* of a belief-base. Moreover, if we seek an epistemic model which is reasonably human-like in its attributions of belief, we shall want to permit distinct formulae to express the same belief. However, the degree of granularity (for equivalent beliefs) which is desirable for an ATMS may well be different from that required for human-like epistemic notions. We shall return to these issues in section 4.

Before leaving this discussion of purely syntactic epistemic models, we should note that Fagin and Halpern (1985) present what might be viewed as a *liberalized* syntactic belief model. The Fagin-Halpern (F-H) model identifies an agent's beliefs with a subset of formulae of which the agent is aware. This particular subset (which satisfies special conditions) *may* be taken to include formulae the agent will only *come to be* aware of at some future time. In this respect the F-H model is less restrictive than those which restrict an agent's beliefs to formulae presently written in the agent's belief base. However, apart from the awareness requirement, the F-H model is not concerned with the empirical preconditions of belief (e.g., that a sophisticated robot might derive a formula from existing beliefs, but not "commit to" the formula until a "consistency check" with other beliefs is complete).

There is no reason to suppose the F-H model was ever intended to deal with the empirical preconditions of belief, but an agent who wishes to reason (or conjecture) about what a given robot *currently* believes will require a model of belief which makes explicit many of these preconditions. Such a model will need to recognize the fact that agents revise and retract beliefs, and that they sometimes derive formulae which they *never* accept, even though the derived formula is indeed a valid consequence of the agent's existing beliefs.

1.3 The Intensional Approach

The intensional approach is founded upon the premise, widespread among philosophers of language, that belief and knowledge must be understood as a relation between an agent and an abstract *intension* (proposition or thought). Recent work in epistemic modeling, which adopts an intensional stance,

includes that of Wilks and Ballim (1987), who are concerned primarily with the *generation* of nested beliefs (especially atypical beliefs). By contrast, our primary concern is not with the *generation* of nested beliefs, but with adequate methods of *representing* nested epistemic states (including beliefs involving complex nesting of quantifiers), and with the presentation of a *formal* model which makes *explicit* empirical preconditions which agents must consider when reasoning about each other's belief system. It will be argued that our formal model (NIM) can, given an appropriate understanding of 'intension', yield reasonably human-like notions of belief and knowledge. Moreover, upon a non-standard construal of 'intension', NIM can be seen as a formal theory of the epistemic states of an ATMS. In order to better understand the senses of 'intension' at issue, let us briefly review some of the history of this concept.

The thesis that a declarative sentence expresses an abstract sense or intension, whose structure is a *function* both of the sentence's syntax and of the meaning of its parts, is a view which dates back at least to Frege. Modern philosophical descendents of Frege, including Montague (1970) and Lewis (1976), have considerably refined and modified Frege's view, but Frege's underlying idea that the sense (intension) of a sentence is determined by a compositional process, involving both syntax and meaning functions, persists. The precise definition of 'intension' varies from theory to theory, but a sentence's intension is usually taken to be a *function* which, given a possible world as argument, returns the truth value of the sentence in that possible world.

One notorious difficulty with intensional semantics is that when intensions are merely construed as functions in the set-theoretic sense, then every tautology, ϕ , turns out to have the same intension (or meaning), since each valid ϕ must be assigned the value 'true' in each possible world. (Recall that ϕ 's intension is a function from possible worlds to truth values). Thus, when intensions are regarded *merely* as functions, intensions are too coarse-grained to serve as meanings or "thought contents". In an attempt to circumvent this difficulty, Lewis (1976) defines sentence meaning as a function both of a sentence's *parse tree* and its intension. In doing so, he quickly disposes of the problem concerning tautologies. However, many believe Lewis's modification results in a theory of meaning which is *excessively* fine-grained, since under his modification virtually *every* syntactically distinct pair of sentences (e.g., $p \vee q$ vs. $q \vee p$) is assigned a different meaning. In (Hadley, 1986) a simple modification of Lewis's view is proposed to remove this difficulty. The modification creates a degree of granularity for intensions which allows us to say that *structurally* non-isomorphic formulae (including tautologies) differ in intension, but formulae which are both logically equivalent and structurally isomorphic in a graph-theoretic sense, have the same intension or meaning. For example, on this account $p \vee q$ and $q \vee p$

have the same intension since they are logically equivalent and their parse trees are isomorphic.

Other *compositional* theories of sentence meaning exist, of course, and each has its own degree of granularity. Cresswell [1985] proposes a theory of sentence meaning with an attractive degree of granularity, as does Gawron [1986]. For the present, we may remain neutral among these competing theories. We shall simply identify intensions with sentence meanings, and note that our choice of granularity for intensions could vary according to the intended application of NIM. We return to these issues in section 4. What is relevant for present purposes is that each of the aforementioned theories proposes a *canonical* system for representing meanings or intensions, such that *all sentences which are synonymous* (according to the semantic theory) *are mapped onto a single canonical representation*. Moreover, each of the semantic theories presupposes the existence of a *principled* set of rules for translating (or compiling) an expression into its canonical meaning representation. Questions about synonymy may be answered by comparing the canonical representations into which the respective expressions are "compiled" by the translation rules. Questions whether two distinct expressions can represent the same belief will in turn be decided by ascertaining whether the distinct expressions are synonymous.

Before leaving the present topic, the reader should note that our use of 'synonymy' is somewhat technical. There has been controversy among philosophers regarding the interchangeability of "synonymous" sentences within belief contexts. For example, many philosophers of language would agree that 'John is someone's brother' is synonymous with 'John is someone's male sibling', because the sentences are identical, except that the former contains 'brother' where the latter contains an expression analysing the concept of 'brother'. However, some (including Moore, 1942) have denied that the two aforementioned sentences are interchangeable in belief contexts. In general, the problem of 'concept analysis' is a thorny one in philosophy. Moore was among the first to draw attention to what he labelled the 'paradox of analysis', i.e., that the analysis of a concept is usually not substitutable (in a belief context) for an atomic term expressing that concept. For our purposes, the reader should bear in mind the following: first, the paradox of analysis is controversial. Moore himself concedes that one cannot "come to know" that a concept C applies in a given situation without "coming to know" that the analysis of C holds true in that situation. Thus, one could not come to know that X is a brother without coming to know that X is a male sibling (though one may not have words to express this knowledge). This certainly suggests that there is a *sense* of belief in which substitutivity holds, and in particular applications we may be concerned with this sense of belief. Also, Barnden (1986) makes a convincing case that our willingness to count two

expressions as "interchangeable in belief contexts" is to some degree a function of our present interests. Second, if our present concerns require that we regard 'X is a brother' and 'X is a male sibling' as *not* interchangeable in belief contexts, we can find a sufficiently fine-grained criterion of 'intensional identity' to yield this result. For example, the aforementioned requirement that "parse trees be isomorphic" ensures the non-interchangeability of these expressions while still allowing us to say that 'tadpoles swim' and 'pollywogs swim' are synonymous (in our technical sense of synonymy).

1.4 The Proposed Model

In what follows an axiomatic system is presented which aims to formalize many of the interrelationships among the epistemic concepts, and related notions such as retracting, questioning, or committing to a belief. The model has the *capacity* to represent human-like epistemic concepts, and is intended to be of use to those who seek *formal* models of epistemic *processes*. Our overriding concern is not with finding a tractable model, but with providing a conceptual map which is sensitive to the empirical contingencies of belief and knowledge.⁵ The epistemic model to be described is axiomatized in first-order logic. Self-reference is possible within the language of the model; thus, the potential for paradoxes of self-reference arises. A method (due to Kripke, 1975) for eliminating the contradictions which self-reference can engender is described in section 2.3.

In sections 2.1 and 2.5 the representation language and axioms of the formal model are presented. The axioms strongly modify and generalize the *intensional* model of belief proposed in (Hadley, 1988) (which included neither *knowledge*, the representation of nested epistemic propositions, nor self-referential belief statements). As will become apparent, the model proposed here ('NIM' for "nested intensional model") requires an agent to know or believe *only* those formulae which are intensionally identical with formulae explicitly present in the agent's belief base. Thus, NIM does not entail logical omniscience. Moreover, NIM does not require us to suppose that agents believe every formula they have validly *derived* from other beliefs.

2. AN INTENSIONAL EPISTEMIC MODEL

2.1 The Language of Nim

The axiom set to be presented in section 2.5 permits agents to reason about their own epistemic states and

⁵There is no reason to suppose that all relevant aspects of belief and knowledge can be expressed in tractable formal systems. We know, for example, that many forms of mathematical reasoning are intractable, and the resultant theorems are not expressible in a tractable logic. In addition, many planning problems appear insoluble, except by means of intractable search processes. If artificial agents are to engage in *hard* planning of this kind, it seems clear that they will require intractable logics. Also, if these agents are to formulate beliefs *about* how their plans were derived, it is difficult to see how these beliefs could be *represented* except in an intractable logic.

those of others. Thus, the language of our model includes nested (embedded) epistemic formulae. However, the language should also permit us to say that different agents believe different formulae at different times. For this reason, epistemic assertions should involve at least three separate parameters. Now, the representation of assertions involving multiple parameters is straightforward in first-order logic (FOL), and for this reason (among others) FOL is chosen. However, the representation of *nested* epistemic propositions in FOL is less straightforward.

Perlis (1985) describes a method, involving quotation, unquotation, and concatenation of expressions, which permits the representation of nested propositions in an extension of FOL. Haas (1986) presents a quotation-based scheme which is, in many respects, analogous to that of Perlis. However, both the methods of Perlis and Haas become unwieldy when an epistemic assertion is nested in an epistemic context. For example, in the sentence below,

- (1) John knows that Mary knows that John knows Mary's phone number.

the most deeply nested sentence, "John knows Mary's phone number" must be *true*, since it is known to be known, and whatever is known is true. We wish our representation scheme to facilitate the inference of "John knows Mary's phone number" from (1). To enable this inference, Perlis must ensure that each sentence at each level of embedding is "detachable", so that when (1) is combined with an axiom such as (2) (below) the embedded implications may be extracted.

- (2) If an agent knows sentence Y, then Y is true.

Thus, in Perlis' scheme (1) is represented as follows:

- (3) *Knows*(John, "*Knows*(Mary, " $\exists x$ [*digits*(*x*) $\wedge$ *Knows*(John, *concat*(*x*, "*=number*(Mary))")])")")

In (3), 'concat' expresses a concatenation function which returns a quoted expression, *after* constant values have been assigned to the variable parameters of 'concat'. Thus, if *x* were bound to '7345727', 'concat' would return the string "7345727 = number(Mary)". In Haas' system, (1) would receive a different, but equally complex representation. By contrast, the representation scheme in NIM involves no quotation mechanism. NIM relies instead on the *composition of functions* to ensure both that the objects of belief are intensions, and that inferences are not unduly encumbered by an plethora of quotation marks.

For example, in NIM (1) would be represented as⁶

(4) Knows(john, Knows(mary, $\exists x$ (Knows(john, eq(x, phone_num(mary))))))

Now, at first glance (4) appears not to be a sentence of first-order logic (FOL). However, as we shall see, if we allow the role of predicates, quantifiers, and truth connectives to be *context sensitive*, so that they express functions in any embedded context, then (4) becomes a sentence of FOL. The underlying motivation for this approach is not merely to simplify our system of representation and inference (though we do obtain those benefits). Rather, we are attempting to embody Frege's thesis that the denotation of a sentence (or wff) varies according to its surrounding context. Frege maintained that sentences which occur in embedded epistemic contexts (as in (1)) denote their *sense*, and not their truth value. As previously mentioned, this *sense* (intension) is formed through the composition of functions. Additional restrictions may be placed upon intensions to obtain differing degrees of granularity, but all intensional theories relevant to our discussion posit that intensions are formed through the composition of functions. In some respects, what we are proposing is similar to an n -order intensional logic (where n is unbounded). However, every axiom of NIM, and all formulae which we shall need to represent, are expressible in FOL. We shall be extending standard FOL by the inclusion of two rules of inference (which concern the predicate 'true'), but these do not affect the representation *language* (hereafter, $\mathcal{L}$).

In order to implement Frege's thesis concerning the context-sensitivity of denotations, we adopt the following notational conventions: the predicates T (for true), K (for knowledge), and B (for belief) are distinguished predicates, and always appear in upper case; other predicates, constants, and variables are written in lower case. Officially, the truth-functional connectives, ' $\neg$ ', ' $\rightarrow$ ', etc., are *prefix* connectives whose arguments are written within parentheses. Thus, we write $\rightarrow (\forall x (pred1(x)), \exists y (pred2(y)))$. Also, any time a quantifier is introduced, the top-level formula within its scope is enclosed in parentheses, as in $\exists x (pred2(x))$. In practice, we shall omit parentheses where the scope is unambiguous, and use connectives in their customary infix position. Our official notation is intended merely to stress that quantifiers and connectives may be viewed as functors. Of course, predicates are commonly regarded as functors by modern logicians; thus ' $red(x)$ ' is usually called a propositional function.

⁶For now, we ignore the *times* at which various propositions are known. In NIM, temporal parameters are included. They are omitted here in order to obtain something comparable to Perlis' formulation. The temporal aspect of belief is ignored by Perlis, but included by Haas. It is worth noting that propositions such as (1) above, are not expressible in Maida's (1976) formalism for nested belief statements. For, apart from the (previously described) restrictions inherent in purely syntactic approaches, Maida's *propositional* representations cannot represent the *internal* quantificational structure of propositions (as Maida concedes).

Now, when a predication (say, $red(x)$) is embedded in an epistemic context, the predicate involved (red) expresses a function, $f1$, whose parameter is x , and whose value is a proposition whose identity varies according to x . Thus, when x is assigned the value a , the function $f1$ returns, *but does not evaluate*, a proposition which is the standard, non-embedded intension of ' $red(a)$ '. This intension is a function, $f2_a$, which assigns different truth values to ' $red(a)$ ' in different possible worlds. Analogously, when a truth-connective, (say, ' $\rightarrow$ ') appears in an embedded context (as in $K(john, \rightarrow(red(x), round(x)))$), it expresses a function, $f3$, which takes as arguments whatever propositions (intensions) are returned by $red(x)$ and $round(x)$, and $f3$ returns a new function, $f4$, which is simply the standard truth function attached to $\rightarrow(red(x), round(x))$ in FOL. Of course, $f3$ cannot actually return a given intension until a value is assigned to x . When a value (say, a) is assigned to x , $f3$ returns ($f4_a$) the intension of $\rightarrow(red(a), round(a))$. Again, $f3$ merely returns, but does not evaluate $f4_a$.

The behavior of quantifiers in embedded contexts, although similar in some respects to the above, differs in important respects. To see this, let $f5$ be the function expressed by ' $\forall x$ ' in an embedded context, and let $g1$ be the function expressed by ' $round$ '. Then consider the general form of $K(john, \forall x \{ \rightarrow(red(x), round(x)) \}, time-c)$ when *embedded* quantifiers, connectives, and predicates are replaced by the function names we have introduced. Performing the replacement gives us:

$$(3) K(john, f5\{f3(f1(x), g1(x))\}, time-c)$$

Given that $f5, f3, f1$ and $g1$ are all functions, $f5$ (which is expressed by ' $\forall x$ ' only in embedded contexts) ultimately returns an intension by means of straightforward composition of functions. Now, at first glance, it may appear that occurrences of ' x ' in $f5\{f3(f1(x), g1(x))\}$ are free occurrences, since $f5$ is just the outer-most function occurring in the atomic formula (3). This appearance is illusory, however. To be sure, occurrences of ' x ' in $f3(f1(x), g1(x))$ are free, but when this latter expression is supplied as an argument to $f5$, the value returned by $f5$ is the ordinary *non-embedded* intension of $\forall x \{ \rightarrow(red(x), round(x)) \}$, which does not contain free occurrences of ' x '. This is because $f5$ is a function expressed by an *embedded* quantifier of ' x '. As such, $f5$ expects to be passed a (propositional) function involving free occurrences of ' x '. So, although $f3$ cannot return a value until ' x ' is bound to a constant, $f5$ can return a value as soon as it is passed the open function $f3(...x...)$. (If we were working in LISP, $f5$ would be an *nlambda* function, because $f5$ does not attempt to evaluate its argument before using it.) What $f5$ returns is another function ($f6$), but $f6$ is simply the sense of a closed, quantified sentence of

$\mathcal{L}$.⁷ Now, if the argument supplied to $f\beta$ was a function containing free occurrences of another variable 'y', as well as 'x', then the value returned by $f\beta$ would still be an open function -- but only open with respect to 'y' -- occurrences of 'x' would not be free.

By contrast, consider the sentence, $\exists x (B(\text{john}, \text{blue}(x), \text{time}-c))$. The predicate 'blue' in this embedded context expresses a function which returns particular propositions for given values of 'x'. Since no *embedded* quantifier binds occurrences of 'x' here, closure of $B(\text{john}, \dots)$ requires the prefixing of a quantifier (in this case, $\exists x$).

2.2 Truth in NIM and in Perlis

Hopefully, it is now clear that the functions we have chosen to assign to predicates and logical operators -- *when they occur in embedded contexts* -- are such that when a first-order sentence, ψ , is nested inside another wff, ψ *returns* (but does not evaluate) the function which *constitutes* the ordinary, non-embedded sense of ψ . For this reason, we may justify the addition of two rules of inference to our first-order theory, given the convention just described. These rules allow us to move an expression into or out of an embedded context of the following form:

(4) $T(\phi)$.

Informally, we may construe T as the "truth" predicate. Thus, (4) would assert that the *intension* denoted by ϕ is 'true'. A more formal interpretation for T is provided in section 4. At this stage, it is important to note that we do not attempt to *explicitly* define the truth predicate within the *language* of NIM. However, the following two rules of inference express the *logic* of this predicate, and they are included in our extension of FOL.

T1: ϕ may be derived from $T(\phi)$, provided ϕ has the typographic form of a (closed) sentence of $\mathcal{L}$.

T2: $T(\phi)$ may be derived from ϕ .

Note that T1 allows us to "extract" *functional* expressions from a truth context, under certain conditions, and T2 allows us to *embed* a sentence in a truth context. Now, there are circumstances in which we wish to assert a sentence such as ' $T(c)$ ', where ' c ' is an individual constant denoting some intension. In such cases, T1 does *not* allow us to derive c as an assertion, since c is not a sentence of $\mathcal{L}$.

⁷Although $f\beta$ is not evaluated in this embedded context, $f\beta$ would return truth just in case, for every value assignable to x in the present possible world, $f\beta$ yields a proposition which *would* evaluate to true. From a computational standpoint, $f\beta$ is able to iteratively invoke $f\beta$ (with different values for x), because the lambda function, $f\beta$, made $f\beta$ a component of $f\beta$.

Insofar as T1 and T2 allow us to move formulae into and out of a truth context, their joint effect bears some resemblance to Perlis' axiom schema:

$$(5) \text{ True}("A") \equiv A*$$

where A is a meta-variable ranging over sentences of the language, and $*$ is an operator which ensures that (5) is equivalent to

$$(6) \text{ True}("A") \equiv A$$

in those cases where ' A ' does not contain the predicate 'True'. In effect, the $*$ operator is a device which blocks the derivation of $\text{True}("L")$ from L in cases where $L \equiv \neg\text{True}("L")$, i.e., in cases where L is the famous "Liar" sentence. In such cases, Perlis allows that L and $\neg\text{True}("L")$ may both be *proven*, while $\text{True}("L")$ may not be derived. Thus, he avoids explicit contradiction, but in the process he adopts a sense of 'truth' which may do violence to our ordinary concept of truth. For, it seems counter-intuitive to say that we are *sometimes* perfectly entitled on semantic grounds to assert a sentence (as when, for example, L has been *proven*), but we are not entitled on those occasions to say the sentence is true. Kripke's view, which many would share, is that a correct theory of truth ought to preserve the following common usage: we say that a sentence is true in *exactly* those situations where we are semantically entitled to simply assert the sentence. (Note that NIM's rules, T1 and T2, preserve this common usage. *Whenever* we have justifiably asserted a sentence B , T2 allows us to infer " $\text{True } B$ ". T1 sanctions the reverse inference for any sentence.)

Apart from the above difficulty, it seems clear that Perlis' use of the $*$ operator (in (5)) cannot, in general, accomplish what he wants it to. To see this, we need to have in mind Kripke's notion of a *grounded* sentence. In essence, a grounded sentence is one whose truth evaluation "bottoms out", either immediately or ultimately, in the evaluation of sentences which do not ascribe truth or falsity to other sentences. By contrast, the truth evaluation of an *ungrounded* sentence does not "bottom out", but triggers an endless series of further attempts to evaluate the truth of itself and/or other sentences. (For more on how truth evaluation can "ultimately bottom out", see Hadley, 1989.)

Now, Perlis clearly intends that the "star" operator will block the derivation of $\text{True}("A")$ from A when A is not grounded.⁸ In Perlis' schema, the "star" operator works in a purely syntactic fashion; in essence, it rewrites its argument by replacing occurrences of $\neg\text{True}("...")$ by occurrences of $\text{True}("\neg(...)")$. But the question whether a sentence is grounded *sometimes* depends upon purely empirical considerations, and

⁸Indeed, Perlis must hold this if (5) is to prevent 'True' from being ascribed to the subtle kinds of paradoxical sentences which Kripke discusses (cf. Kripke, 1975, and Perlis, 1985, pp. 310-311).

cannot reliably be determined by syntactic means. This was a major conclusion of Kripke's paper. Given Kripke's conclusion (which Perlis does not dispute), it follows that the star operator cannot do both the following things: (a) always block the derivation of $True("A")$ from A , when A is ungrounded, (b) always permit (as Perlis claims) the derivation of a sentence having the form $True(...)$, from A , when A is grounded. Since the same sentence can be grounded in some possible worlds, and not in others, no purely syntactic device can reliably distinguish between semantically valid and invalid derivations of $True(...)$ from ungrounded A . As it happens, schema (5) works reliably only for *explicitly* self-referential sentences, such as the Liar.

2.3 Paradox and Truth in NIM

Let us now consider whether the paradoxes of self-reference can arise in NIM. To begin, we observe that the explicit liar paradox cannot be formulated in NIM. To be sure, we can represent *instances* of the following within the language of NIM ($\mathcal{L}$):

$$(7) \phi \equiv \neg T(\phi)$$

by replacing ϕ with any sentence of $\mathcal{L}$, but all *instances* of (7) are simply *false*, not paradoxical. (Within $\mathcal{L}$, only a well-formed, closed formula could be substituted for ϕ in (7). Because there is no convention within $\mathcal{L}$ by which a closed formula could *denote* itself, it is not possible to find a value for ϕ which renders (7) paradoxical in $\mathcal{L}$.) From our standpoint, (7) is merely an absurd meta-schema, and is not expressible in $\mathcal{L}$. Moreover, because rules T1 and T2 are *sound* (cf. section 5), we cannot use rules T1 and T2 to derive an instance of (7) from a consistent premise set. Now, *if* we adopted as a premise, $\neg T(c)$, and *if* the constant c were allowed to name the intension of this premise, then we could demonstrate in our meta-language that $\neg T(c)$ is semantically paradoxical. However, we could not derive a corresponding contradiction within NIM, because

$$(8) c \equiv \neg T(c)$$

is not even well-formed. Moreover, even at the meta-level, we cannot coherently suppose that a constant names the intension of an expression (ψ) in which that constant occurs, because the intension (of ψ) can *exist* only if we have *already* assigned an intension to the given constant. This follows from the compositionality of intensions.

Now, although there appears to be no way in which an explicit liar paradox can be expressed in $\mathcal{L}$, this fact does not preclude the more subtle kinds of paradox described by Kripke (1975). These depend upon exceedingly rare (and arguably, farfetched) empirical conditions arising when each of a pair of sentences refers to the truth of the other. It is doubtful whether the required empirical conditions will arise in fact.

However, we should consider whether the mere possibility that semantic paradoxes will arise renders NIM defective.

To begin, let us note that Kripke clearly demonstrates (1975) that *any* language rich enough to permit the ascription of truth to sentences is vulnerable to paradox. He further demonstrates that this vulnerability in no way presupposes syntactic or semantic ill-formedness. Now, it is true that nearly all logics current in AI escape this vulnerability, but only because they do not sanction inferences involving truth ascriptions. For many specialized applications, one can safely ignore the logic of truth ascriptions, but if we are ever to have an adequate model of *human cognition*, we will need to represent and reason with propositions such as, "What Mary believes is true".

Since we are ultimately concerned with adequate cognitive models, we have included a truth predicate. Thus, paradoxical sentences may arise. However, Kripke (1975) has shown that the inclusion of such sentences need not render a first-order language inconsistent. Kripke proposes a complex theory of truth according to which paradoxical sentences are simply not assigned truth values. The underlying intuition of Kripke's theory is that any ascription of truth to a sentence involves a *presupposition* that the sentence is grounded. On the rare occasions when the presupposition of groundedness is unsatisfied no truth value is assigned. Kripke goes on to argue (following Kleene, 1952) that the existence of sentences without truth values does not require that we abandon any of the usual laws of logic. For example, since " $p \vee \neg p$ " still holds for all cases when p has a truth value, we may retain the law. When p lacks a truth value, the disjunction is *not false*; rather it also lacks a truth value.

The question now arises, must the interpretation of T , the truth predicate in NIM, involve considerations of *groundedness*? In the final analysis, the answer is yes. On the face of it, the interpretation of T is straightforward. We merely assign as the extension of T the set of all intensions which evaluate to 'true', where 'true' is an object in the domain of $\mathcal{L}$. However, upon reflection, we note that the relevant "set of true intensions" includes not only intensions of sentences taken from an agent's representation language (whatever that may be), but also intensions belonging to some sentences of $\mathcal{L}$. Now, some of the latter intensions arise from sentences containing the predicate T . Thus, it becomes problematic that *these* latter intensions should enter into the definition of the *extension* of T (because, from an ontological standpoint, the intension and extension of a term arise together). The problem is to prevent a vicious circularity from entering into our specification of the "set of true intensions" (which we denote by ' S ').

Our solution, in its essentials, derives from Kripke (1975). We define the membership of S *recursively*, as

follows: (In what follows, assume that ϕ is a sentence in the domain of $\mathcal{L}$. The domain of $\mathcal{L}$ includes sentences and intensions belonging to the representation languages of the agents concerned, and also *all* sentences and intensions belonging to $\mathcal{L}$ itself.)

- (i) If ϕ contains no occurrences of T , and if the intension of ϕ returns 'true' on the current interpretation of ϕ , then the intension of ϕ is in S .
- (ii) Otherwise, if ϕ is a sentence of $\mathcal{L}$ having the form $T(\psi)$, then the intension of ϕ is that function f which returns 'true' iff ψ is both grounded and the intension of ψ evaluates to 'true'. If f returns 'true' on the current interpretation of $\mathcal{L}$, then $f \in S$.
- (iii) Otherwise, ϕ is a molecular or quantified sentence of $\mathcal{L}$, and contains some occurrence of T . The intension of ϕ (again, call it f) is formed according to the standard compositional rules of Tarskian semantics, and $f \in S$ iff f returns 'true' on the current interpretation of $\mathcal{L}$. Since the subexpressions of ϕ are recursively evaluated for truth in the normal Tarskian fashion, the evaluation of f ultimately reduces to conditions (i) and (ii), which are straightforward.

Given the above definition, the membership of S is fully specified. This, in turn, provides a determinate extension for the predicate T . The fact that T 's extension is recursively defined is no more problematic than if we had assigned the natural numbers as T 's extension. As it happens, however, the chosen interpretation assures the soundness of rules of inference, T1 and T2, as we see in section 2.6.

2.4 Background Assumptions for NIM

Before presenting the axioms of NIM, it may prove helpful to make explicit a few more background assumptions. In the preceding discussion we have assumed that the objects of belief, knowledge, and truth are intensions, and that intensions are functions. We also assume, however, that *each* such intension will be represented by a canonical expression. As previously indicated, we are not committed to a particular method of representing intensions, but we shall nevertheless *identify* intensions with their canonical representations (regardless of how we choose to represent intensions). We also assume that each agent possesses a belief-base (BB) which contains a set of *nodes*, and that for each *believed* intension there exists some node in the BB which stores some formula f , such that f explicitly (or conjunctively) expresses the believed intension in the agent's own representation language. (We say that f conjunctively expresses a given intension if f is a conjunction, one of whose conjuncts expresses that intension.) Formula f need not be *our* canonical representation of the believed intension; indeed, no *canonical* representations of intensions need be explicitly stored in an agent's BB.

We shall later argue that for some applications it is computationally advantageous to store, explicitly, every intension which is *expressed* by a formula in the BB.⁹ Intensions which *are* explicitly stored

⁹Hereafter, when we say 'intension', we shall mean 'canonical representation of intension', unless the context indicates otherwise.

should be located at every node which contains any formula expressing that intension. When an intension is *not explicitly* stored, we shall still consider it to be in the BB, provided some formula expressing that intension is explicitly present in the BB. When several formulae are synonymous (i.e., have the same intension), it is desirable (for complexity reasons) that they all be stored at the same node. At present, however, we leave it open whether synonymous formulae *must* be stored at the same node.

Thus far, we assume that a BB will contain certain explicit formulae (sentences) and *may* contain explicit representations of intensions corresponding to those formulae. Related to this is the fact that we sometimes speak of beliefs as though they were *sentences*, and at other times as though they were the *thoughts* expressed by sentences. In order to reflect this disposition in our speech, and to accommodate the fact that our logics deal with written formulae, we shall hereafter assume that the *extensions* of *B* (belief), *K* (knowledge), and *T* (truth) include both intensions and all sentences which compile into those intensions. For example, the extension of *T* will be the set of intensions which return true (on a given interpretation) together with all sentences which compile into any of those intensions. Likewise, the extension of the predicate *B* (as in $B(\text{agent}, \phi, \text{time})$) will be the set of all intensions the agent believes together with those formulae which express (compile into) those intensions. Thus, we may obtain true instances of (10)

$$(10) \quad K(a, f, t) \rightarrow B(a, f, t) \wedge T(f)$$

either by replacing *f* with a wff of $\mathcal{L}$ (which in the *embedded* context must denote an intension), or by replacing *f* with a constant, say *cI*, whose denotation may be either an intension or a sentence. (In (10) the variable '*a*' stands for agent, and '*t*' for time. Unless otherwise indicated, assume variables are universally quantified with widest possible scoping.)

In addition to the foregoing, we assume that any *axiom* which the agent accepts must be represented in the BB by at least one formula. Beliefs which are *caused* in the agent via processes not normally considered inferential (e.g., via sense perception) may be regarded, for present purposes, as axioms acquired at various times. In addition, the BB may contain formulae which the agent has explicitly inferred and "accepted". We assume that all formulae in a BB belong to an artificial language, but we make no assumptions about the particular language and inference rules which the agent *X* employs. If a particular agent lacks the language and axioms of NIM, we may still use NIM to reason *about* that agent's epistemic processes. Of course, an agent who possesses NIM may do the same. We do *not* assume that *X*'s axioms form a consistent set, or that *X* never makes a fallacious inference.

Finally, we assume that agents *may* employ strategies for maintaining consistency, and may, as a consequence, retract previously held beliefs. The retraction process may, or may not be ideally rational. We do not assume that an agent enters a sentence into his BB as soon as it is inferred. There is a sense of 'infer' according to which one believes anything one has inferred, but here we use 'infer' to mean merely 'derive according to accepted rules (which may include heuristics)'. Agents may treat newly inferred formulae as "candidates" for belief, and enter an examination or "questioning" phase during which the candidate is tested for consistency with all or some of the agent's other beliefs.

2.5 Axioms of the Model

We now present the axioms of NIM. As will be apparent, NIM contains an unusual number of predicates whose interpretation is underconstrained by the formal structure. This is due, we believe, to the fact that it deals with *aspects* of belief which are simply ignored in most *axiomatic* treatments. Our contention is that one simply cannot form reasonable conjectures about an agent's belief states over time, unless these additional aspects are explicitly considered. For clarity, the intended interpretations of each axiom and predicate are discussed as the axioms are introduced.

The expression $B(i, f, t)$, which occurs in most of the axioms, may be interpreted as *agent i believes f at time t*. The predicate $in-bb(i, f, t)$ may be rendered: *f is explicitly present in i's BB at time t*.

$$A1. \quad in-bb(i, f, t) \rightarrow B(i, f, t)$$

Axiom A1 tells us that all formulae and intensions explicitly present in *i*'s BB at time *t* are among *i*'s beliefs at *t*. In axioms A2 and A3, below, the term '*sense(f)*' expresses a function which returns the *intension* of formula *f*. Thus, axiom A2 states that a formula is believed iff its intension is believed, and iff all synonymous formulae are believed.¹⁰ If we assume that formulae *f1* and *f2* are synonymous, and that *f1* is believed, then axiom A2 permits us to infer that *f1*'s intension is also believed, whether or not that intension is explicitly stored in *i*'s BB. Then, given that *f1*'s intension is believed, axiom A2 allows us to infer that *f2* is also believed, whether or not *f2* is explicitly stored. Axiom 3 makes an analogous statement about knowledge. Recall that, for artificial languages at least, the synonymy of formulae may be ascertained by compiling the formulae into their canonical intensional representations, and checking the results for identity.

$$A2. \quad [formula(f1) \wedge formula(f2) \wedge identical(sense(f1), sense(f2))]$$

¹⁰Please recall that the criteria for synonymy can be chosen to ensure that, for the application in mind, synonymous formulae are interchangeable in belief contexts. Also, when we say that a formula is believed, we mean that *what the formula asserts* is believed. An agent may believe what a formula asserts without understanding the formula itself. Thus, on our usage, John may believe what 'Mary has a sibling' asserts, even though John does not understand the word 'sibling'.

$$\rightarrow \{ [B(i, f1, t) \equiv B(i, \text{sense}(f1), t)] \wedge [B(i, \text{sense}(f1), t) \equiv B(i, \text{sense}(f2), t)] \}$$

$$\begin{aligned} \text{A3. } & [\text{formula}(f1) \wedge \text{formula}(f2) \wedge \text{identical}(\text{sense}(f1), \text{sense}(f2))] \\ & \rightarrow \{ [K(i, f1, t) \equiv K(i, \text{sense}(f1), t)] \wedge [K(i, \text{sense}(f1), t) \equiv K(i, \text{sense}(f2), t)] \} \end{aligned}$$

Axiom A4, below, expresses the thesis that whoever believes a conjunction must also believe its constituent conjuncts. At first glance, this thesis may seem problematic. However, recall that we identify 'believing a formula' with 'believing what the formula asserts'. It is very difficult to see how one could believe what a conjunction *asserts* without believing each of its separate conjuncts. One could, of course, believe (entirely on authority) that a formula (in a textbook, say) is true, without even comprehending the formula, but *that* would not qualify as believing what the formula asserts.

$$\text{A4 } [B(i, f, t) \wedge \text{conjunction}(f) \wedge \text{has-as-conjunct}(f, \text{conjunct})] \rightarrow B(i, \text{conjunct}, t)$$

The following axiom expresses the widely accepted view that whoever knows *f* must at least believe *f*, and *f* must be true.

$$\text{A5. } K(i, f, t) \rightarrow B(i, f, t) \wedge T(f)$$

Note that, if we assume $K(\text{mary}, K(\text{john}, \exists y(P(y)), t1), t2)$, then we may use axiom A5, together with conjunctive simplification, to derive $T(K(\text{john}, \exists y(P(y)), t1))$. Rule T1 then allows us to obtain $K(\text{john}, \exists y(P(y)), t1)$, which in turn leads to $T(\exists y(P(y)))$, via axiom A5. We may now use T1 again to derive that $\exists y(P(y))$ as a consequence of our original assumption.

Axiom A6 simply states that an agent who retracts a formula ceases to believe the formula.¹¹ This, together with axiom A5, entails that the formula would not be known. We may use axiom A2 to show that the agent would also cease to believe all synonymous formulae. Axioms A1 and A6 together entail that a retracted formula ceases to be in the belief base.

$$\text{A6. } \text{retracts}(i, f, t) \rightarrow \neg B(i, f, t)$$

In the following axiom, *questioning*(*i*, *f*, *t*) may be interpreted as *i begins at time t to investigate whether formula f is consistent with other beliefs*. The underlying idea is that, if *i* has begun to question whether *f* should be retained, then *i* is not presently committed to *f*'s truth. We assume that those automated reasoning systems which employ belief revision strategies enter a phase in which the consistency of a set

¹¹In our discussion of the remaining axioms, A6-A12, we speak of the variable *f* as ranging over formulae. However, since axioms A2 and A3 ensure that whenever a formulae is believed its corresponding intension is believed, and since we are now treating intensions as special canonical formulae, we omit explicit reference to intensions throughout the remainder of this section.

of formulae is investigated before any retractions are made. A formula which is being questioned may be one which has been singled out for special investigation. However, we do not insist that 'retracts' and 'questioning' be interpreted just as suggested here, and no attempt is made to formalize these notions. If a given system lacks a questioning mechanism, or even a retraction mechanism, then the 'questioning' predicate (or 'retracts') may be assigned the value 'false', which would render the presence of these predicates innocuous. On the other hand, if the system has a TMS, the TMS will provide advice about which formulae are to be retracted. Thus, implementation of the 'retracts' predicate is comparatively straightforward. By contrast, the *questioning* phase (searching for inconsistencies) is not *only* the task of the TMS, but of an associated problem solver. Thus, implementation of the 'questioning' predicate is not trivial.

$$A7. \text{questioning}(i, f, t) \rightarrow \neg B(i, f, t)$$

The intended interpretation of axiom A8 is that any formula which has just been inferred by an agent, from a subset of that agent's beliefs, is to be regarded as a candidate for belief by that agent. In the following, *just-inferred*(i, f, s, t) may be read *i inferred f from set s at time t*. *Mem*(w, s) may be read as *w is a member of set s*, and *cand*(f, i, t) is rendered as *f is a candidate for i at t*.

$$A8. [\text{just-inferred}(i, f, s, t) \wedge \forall w \{ \text{mem}(w, s) \rightarrow \text{in-bb}(i, w, t) \}] \rightarrow \text{cand}(f, i, t)$$

The following two axioms describe additional ways in which a formula may become a candidate for belief. Axiom A9 asserts that a formula which is *presented* to an agent as true (perhaps *via* sense perception) becomes a candidate for belief by the agent. Since the agent may wish to subject such formulae to a "questioning phase" before committing to belief, axiom A9 is included (cf. the remarks following axiom A12, below).

$$A9. \text{just-presented}(i, f, t) \rightarrow \text{cand}(i, f, t)$$

Axiom A10, below, asserts that a formula which *has* once been believed, but which is *later* being questioned, becomes merely a candidate for belief while being questioned. Axiom A7 ensures that formulae are not believed while being questioned.

$$A10. [B(i, f, t_1) \wedge \text{later}(t_1, t_2) \wedge \text{questioning}(i, f, t_2)] \rightarrow \text{cand}(i, f, t_2)$$

The following axiom tells us that, under normal circumstances, beliefs persist. More specifically, the axiom tells us that if *i* believes a formula at t_1 , then *i* also believes the formula at all later times, unless *i* retracts or questions the formula at some intervening time. Since the interpretation of 'retracts' is left to the user's discretion, 'retracts' may be taken to include some form of spontaneous forgetting or other memory loss. Nothing here forces us to restrict retraction to deliberate actions. In A11, *later*(t_1, t_3) may be read as t_3 is later than t_1 , and *inc-between*(t_1, t_2, t_3) may be read t_2 lies inclusively between t_1 and t_3 .

$$\begin{aligned}
 \text{A11. } & [B(i, f, t_1) \wedge \text{later}(t_1, t_3) \\
 & \wedge \neg \exists t_2 (\text{inc-between}(t_1, t_2, t_3) \wedge [\text{retracts}(i, f, t_2) \vee \text{questioning}(i, f, t_2)])] \\
 & \rightarrow B(i, f, t_3)
 \end{aligned}$$

The preceding eleven axioms are intended to be *descriptive* of our concept of belief. They relate the concept of belief to cognate concepts in a manner intended to be useful to agents who reason about the beliefs of an agent. The following axiom, by contrast, is *prescriptive*. It is intended to convey the conditions under which a candidate formula (in the sense introduced by axiom A8) *should* become an active belief. The term '*successful*(*i, f, t₂*)' should be taken to mean that at time *t₂* formula *f* has just been successfully scrutinized by agent *i*'s questioning mechanism, and has been found acceptable. The term '*should-be-inserted*(*i, f, t₂*)' may be read as, formula *f* should be added to *i*'s BB at time *t₂*. Thus axiom A12 states that a formula which is a candidate for *i* at *t₁*, and which has successfully passed the questioning phase at a later time *t₂* should be added to *i*'s BB at *t₂*. Axiom 1 would then ensure that *f* is a belief at *t₂*.

$$\text{A12. } [\text{cand}(f, i, t_1) \wedge \text{later}(t_1, t_2) \wedge \text{successful}(i, f, t_2)] \rightarrow \text{should-be-inserted}(i, f, t_2)$$

Now, since the interpretation of 'questioning' may vary according to the intended application of the model, the interpretation of 'successful' should vary accordingly. If a particular agent altogether lacks a questioning mechanism, we may simply assign 'true' to 'successful', since there would be no tests which a candidate formula should pass, beyond being a candidate. Of course, questions may arise about how exhaustive a questioning or consistency testing process must be before we may assign 'success' to a formula. However, these are practical issues which must be decided in particular applications.

2.6 Formal Properties of the Model

We now consider some of the formal properties of NIM. First, note that the axioms of NIM, with minor syntactic alterations, may be represented in any standard first-order predicate calculus, known to be sound and complete. We use 'PC' to denote those formalizations of *pure* first-order predicate logic, which do *not* contain the propositional calculus as a subset.¹²

Definition. Let *T* be a first-order theory formed by adding to PC an encoding of the axioms of NIM in PC, together with inference rules T1 and T2. Further assume that whatever Tarskian interpretation is assigned to *T*, the *predicate* *T* is assigned the truth interpretation described in section 2.3.

Proposition 1. *T* is sound.

Proof. The inference rules of *T* are exactly those of PC combined with the truth rules T1 and T2. Since the inference rules of standard PC are known to be sound, we need only demonstrate that T1 and T2 are

¹²We wish to exclude those formalizations of FOL which permit *constants* to stand alone as wffs of the language. Thus, a sentence such as $P \equiv T(P)$ does not occur in any logic denoted here by 'PC'.

also truth preserving. Because both these rules are explicitly concerned with truth, the proof is straightforward. In essence, T1 allows us to derive ϕ from $T(\phi)$, provided ϕ is a wff of $\mathcal{L}$. Now, suppose that $T(\phi)$ is true under an interpretation I , and that ϕ is a wff of $\mathcal{L}$. By definition of T , I assigns the predicate T the *truth* interpretation. Furthermore, our definition of a *truth* interpretation (section 2.3), requires that $T(\phi)$ can be true only if ϕ has a true intension. But, because ϕ is a wff of $\mathcal{L}$, its intension will be true only if ϕ is true under I . Thus, T1 is truth preserving.

To see that T2 is truth preserving, we reverse the foregoing argument. T2 allows us to derive $T(\phi)$ from ϕ . Suppose that ϕ is a sentence of $\mathcal{L}$ which is true under I (as above). Then the intension of ϕ must evaluate to 'true'. Consequently, the intension of ϕ is in the extension of T . Therefore, $T(\phi)$ is true under I .

Proposition 2. T is complete.

The proof of completeness for T is non-trivial. Details are given in Appendix A.

3. NIM and INTER-AGENT REASONING

Throughout our discussion we have emphasized that the acquisition of beliefs by an agent presupposes that particular empirical conditions are satisfied. This fact complicates the process by which one agent may arrive at reasonable conjectures about the belief states of *another* agent. We are rarely in a position to know *with certainty* what inferences another agent will have drawn at some future time, even when we know that agent's present beliefs. For the inferential processes of another agent may be affected by a number of factors, including unforeseen interruptions in the other's thought processes, and "processing loads" which are presently unknown to us. Nevertheless, at least in the case of comparatively simple inferences, we are often in a position to make *plausible assumptions* about another's thought processes, and so to arrive at reasonable conclusions about particular beliefs another will acquire. In what follows we consider how the language and axioms of NIM can help one (artificial) agent to arrive at conclusions about another agent's beliefs.

To begin, we consider a problem posed by Haas (1986). Suppose that a robot (say, Robin) has the following beliefs:

- (11) John knows what Mary's phone number is.
- (12) John knows that Mary's phone number is the same as Bill's.

Then, at first blush, Robin is in a position to conclude

(13) John knows what Bill's phone number is.

Now, note that, strictly speaking, (13) is not a deductive consequence of (11) and (12).¹³ For John may have the knowledge described in (11) and (12), and be wondering what Bill's phone number is, but because John may not have it *presently in mind* that Mary's and Bill's number are the same, John may not yet have made the required (obvious) inference. Moreover, John's thought processes may be interrupted before John has a chance to discover the obvious. Nevertheless, it would be a reasonable assumption on Robin's part that John will make the inference, and that John will not spend time searching for inconsistencies (i.e., enter a "questioning phase"). Thus, Robin could *plausibly* infer that (13) John knows what Bill's phone number is. The problem arises, how do we formally reconstruct Robin's reasoning?

In his solution to this problem, Haas does not address *all* the subtleties mentioned above. Rather, he is concerned to show that Robin could represent John's knowledge in such a way that Robin could discover that the identity of Bill's phone number is entailed by what John knows. Haas argues that if Robin can discover this entailment, Robin may suppose that John would discover it as well, and make the obvious inference. So, Robin's concluding (13) would be justified (by common sense standards).

Now, given certain background assumptions, Haas succeeds in showing that Robin would be justified in concluding (13). However, as previously noted, Haas' approach to belief is purely syntactic, and involves the adoption of an elaborate quotation mechanism. As a consequence of his syntactic approach, Haas is lead to adopt a complex *reflection schema* (involving an infinity of axioms). The explanation of Haas' reflection schema is abstruse, and would take us far afield (cf. Haas, 1986, pp. 263-268). We hope to show that the intensional approach embodied in NIM leads to a comparatively simple and intuitive solution to Robin's problem. (What follows is intended as a rational reconstruction of Robin's reasoning; we do not suggest that a human reasoner would necessarily engage in the reasoning outlined below, but that reasoning similar to the following would *justify* the conclusion of (13)).

To begin, let us consider how we might represent (11), (12) and (13) in $\mathcal{L}$. "John knows what Mary's phone number is" (i.e., (11)) becomes

¹³Some may argue that (13) is a strict consequence of (11) and (12), on a *de re* interpretation of 'knows'. After all, if we interpret (11) as asserting that John is *acquainted* with a particular object (a number), and if (12) is true, then Mary's phone number must in fact be identical to Bill's. In that case, (13) must be true because (13) merely asserts that John is acquainted with the very same object. However, if one accepts this interpretation of 'knows', one should also accept a modification of the argument, in which (12) is replaced by "Mary's phone number is the same as Bill's". For the validity of the *de re* interpretation only requires that the identity hold, not that John knows the identity. However, I submit that most readers would find the modified argument implausible, indicating that a *de re* reading of 'knows' is unnatural in the arguments we have been considering. In what follows we assume a *de dicto* interpretation of knowledge and belief.

$$(14) \quad \exists x K(\text{john}, eq(x, \text{phone-num}(\text{mary})), t_a)$$

where $eq(x, \text{phone-num}(\text{mary}))$ means "x equals the phone number of Mary" and t_a is a constant denoting the time of utterance. "John knows that Mary's phone number is the same as Bill's" becomes

$$(15) \quad K(\text{john}, eq(\text{phone-num}(\text{mary}), \text{phone-num}(\text{bill})), t_a)$$

Finally, "John knows what Bill's phone number is", which is the supposed conclusion, becomes

$$(16) \quad \exists x K(\text{john}, eq(x, \text{phone-num}(\text{bill})), t_a)$$

We have previously noted that the supposed conclusion is not a deductively valid consequence of the given premises, and it seems clear that insofar as an inference from (11) and (12) to (13) is even *plausible*, some (trivial) inference on John's part must be presumed to take place. But John's inference requires at least *some* time, and unless an assumption is made that (11) and (12) are true *before* (13) is true, it is doubtful that we can justify Robin's concluding (13) from the two premises. It is true, of course, that in ordinary speech we often *assume* on the basis of an assertion such as (11), that John has known Mary's phone number for some *interval* of time. But even when we make assumptions such as this, the time interval implicit in (13) must begin at least slightly later than the interval implicit in (11), if Robin's inference is to be justified. We have simplified matters somewhat, by treating time intervals as isolated points of time. But, *whether or not* we adopt this simplification, the time implicit in (13) should be later than the time implicit in (11) and (12). Therefore, we should represent (13) as (17) below, and not as (16).

$$(17) \quad \exists x K(\text{john}, eq(x, \text{phone-num}(\text{bill})), t_c)$$

Our present assumption is that t_c (the current time of utterance) denotes a time, or time interval, later than the t_a mentioned in (14) and (15). Our goal is to demonstrate that Robin could infer (17) from (14) and (15), given some plausible assumptions that Robin may be assumed to hold concerning John. For the sake of generality, we shall pretend that the predicate K in (14)-(17) is replaced by B , since we are taking knowledge to be a special kind of belief. (Nothing crucial depends upon this; we could easily reconstruct the following arguments in terms of knowledge).

Let us assume that Robin believes (14) and (15) and that those sentences are in Robin's BB. Robin's present task is to discover what John could (soon) come to believe on the basis of John's present beliefs. We further assume that Robin realizes that the way to explore the consequences of an agent's beliefs is to

take the agent's beliefs as premises. This entails removing embedded beliefs from their epistemic context. Because Robin is a sophisticated robot, she quickly realizes that the belief embedded in (14) can only be extracted after existential instantiation is performed. She performs the required instantiation and extracts the resulting object of belief:

(18) *eq(c, phone-num(mary))*

Because the above contains a constant introduced by existential instantiation, (18) serves as an incomplete specification of John's belief. Nevertheless, Robin can reason with (18). Likewise, after extracting the embedded belief from (15), Robin arrives at

(19) *eq(phone-num(mary), phone-num(bill))*

Assuming that Robin understands the transitivity of the '*eq*' predicate, she rapidly derives:

(20) *eq(c, phone-num(bill))*

Now, if Robin is to discover that John *could* construct a proof similar to the above, Robin must have *some* introspective abilities -- she must at least be capable of *noticing* when she herself has constructed a proof. We assume Robin has this ability, and that she can detect a trivial proof when she sees one. (For the present, we shall count those derivations as "trivial proofs" which consist of at most two immediate inferences. We shall later address non-trivial derivations.) On the basis of these abilities (which Robin may possess as *procedures*) she arrives at the following observation:

(21) *trivial-consequence(robin, set-of(eq(c, phone-num(mary)), eq(phone-num(mary), phone-num(bill))),
eq(c, phone-num(bill)),
interval_j)*

Notice that (21) has the general form

(22) *trivial-consequence(agent, set-of(x, y), z, interval)*

Intuitively, we may render (22) as "For the agent indicated, *z* is a trivial consequence of the set of premises *x, y*, and the agent can derive *z* from {*x, y*} in the indicated *interval*."¹⁴ It is important to bear in mind that our agent, Robin, arrives at (21) through knowing the semantics of 'trivial-consequence', and through an act of introspection. Thus, (21) is not the *immediate* product of an inference any more than the

¹⁴We may suppose that 'set-of' expresses a function of two arguments. If both arguments are individuals which are not themselves sets, then 'set-of' returns the set of both individuals. If one argument is an individual while the other is a set, then the former is inserted into the latter, and the result is returned. If both arguments are sets, their union is returned. Using 'set-of', we can represent any set of wffs. Thus, the set {*a, b, c*} can be written as *set-of(a, set-of(b, c))*.

observation, 'I have just derived something', is the product of inference. Also, note that the instances of x , y , and z in (21) are embedded wffs of $\mathcal{L}$, which return *intensions* in this embedded context. Thus, (21) describes a relationship between an agent, certain intensions, and a time interval. However, since derivations involve the use of tangible sentences, our *formal* interpretation of instances of (22) will be: "For the given agent, some sentence *expressing* the intension z can be derived, by a trivial proof (and in the specified time interval) from a set of sentences which express the intensions in $\text{set-of}(x, y)$." The fact that (21) identifies particular *intensions*, rather than particular sentences, permits Robin to be neutral with regard to the particular sentences which Robin may suppose will occur in John's analogue to the proof of (20) (as we shall soon see).

To recapitulate: we have now seen how an introspective agent (Robin) may arrive at the conclusion that a particular inference was trivial for herself. We wish to show that Robin could arrive at an analogous conclusion about another agent (John), i.e., that an inference involving the same set of intensions (as are involved in Robin's inference) would be a trivial inference for John. In our own experience we find that what is truly a trivial inference for us (e.g., *modus ponens* or the transitivity of equality) is a trivial inference for anyone of even approximately equal intelligence. This is an *inductive* observation which most humans make either consciously or tacitly. We make this inductive generalization without knowing the particular reasoning mechanisms (or logics) others possess, and without knowing whether others represent facts to themselves in the *particular* way we ourselves do. If our robot is to engage in the kind of analogical reasoning humans perform, she should have the benefit of the kind of inductive knowledge humans possess. We will suppose, therefore, that Robin has among her axioms the following:

- A13 $[\text{equal-abilities}(\text{agent1}, \text{agent2}) \wedge \text{trivial-consequence}(\text{agent1}, \text{set-of}(x, y), z, \text{interval})]$
 $\rightarrow \text{trivial-consequence}(\text{agent2}, \text{set-of}(x, y), z, \text{interval})$

Axiom A13 simply states that when two agents have (roughly) equal deductive powers, and the first agent can derive a given consequence (by trivial means in a given interval), then the second agent can do likewise. Now, A13 can be used to infer that an inference is trivial for another agent, but for Robin's purposes, this will not suffice. It is not enough that Robin's inference be a trivial one for John; Robin must be in a position to assume that John *has performed* the inference, however trivial it may be. As we have noted, even trivial inferences can be overlooked if an agent does not have all the required premises *presently in mind*. This becomes apparent when we consider slightly less trivial inferences, such as *modus ponens*. One way in which we can be reasonably certain that another agent has a set of premises "presently in mind" is if we have recently discussed with the agent the content of those premises. Indeed, it is in contexts such as these that we commonly infer that another agent has acquired a belief *via*

inference. When we *are* in a position to assume that another agent has had *all* of a set of premises in mind, and we believe the agent can derive a trivial consequence (from those premises) in a given interval, then it is reasonable to assume that the agent will have derived the consequence after the given interval. We may express this generalization within $\mathcal{L}$ as follows:

$$\begin{aligned} \text{A14} \quad & [B(\text{agent}, x, t1) \wedge B(\text{agent}, y, t1) \wedge \text{presently-in-mind}(\text{agent}, \text{set-of}(x, y), t1) \\ & \wedge \text{trivial-consequence}(\text{agent}, \text{set-of}(x, y), z, \text{interval}) \wedge \text{after}(t1, \text{interval}, t2)] \\ & \rightarrow \text{reasonable-assumption}(\text{just-inferred}(\text{agent}, z, \text{set-of}(x, y), t2)) \end{aligned}$$

Note that the consequent of A14 would only be useful to agents who know how to reason with "reasonable assumptions". A reasonable assumption is one which the agent regards as *likely*, but not certain. To use the consequent of A14, Robin would need to realize that the embedded wff, *just-inferred*(.....) should be taken as an *uncertain* premise, which may be used in further inferences, *provided* the uncertainty is propagated. We assume that Robin has some mechanism for propagating uncertainties.¹⁵ We shall also assume that Robin believes that John has had both (18) and (19) *presently in mind* (these are the premises of the inference to be projected).

Now, given A13, and A14, together with the axioms of NIM and Robin's earlier conclusion, (21), we are in a position to show how Robin may arrive at the *plausible* conclusion that John knows what Bill's phone number is. For convenience, let us represent (21) in a shorthand fashion as:

$$(23) \text{ trivial-consequence}(\text{robin}, \text{set-of}(eq1, eq2), eq3, \text{interval}_j)$$

In (23), *eq1* is shorthand for *eq(c, phone-num(mary))*, *eq2* for *eq(phone-num(mary), phone-num(bill))* and *eq3* for *eq(c, phone-num(bill))*. Although the plan of our proof is no doubt obvious, the proof requires some empirical assumptions on Robin's part which should be brought to light. To begin, Robin will want to use A13. To do so, Robin needs to believe that John's deductive abilities are about equal to her own. Let us assume that Robin has such a belief, perhaps acquired through observation. So, in addition to (23) (and other beliefs) Robin's BB contains

$$(24) \text{ equal-abilities}(\text{robin}, \text{john})$$

Using (23), (24) and A13 Robin infers

¹⁵Various systems for propagating uncertainties have been proposed, primarily default and probability based systems. Although we need not commit to any *particular* such system, the qualitative (non-numerical) probability system of Aleliunis (1988) would be suitable.

(25) *trivial-consequence*(john, *set-off*(eq1, eq2), eq3, interval_j)

Now, Robin's BB also includes

(14') *B*(john, eq(c, phone-num(mary)), t_a)

and

(15') *B*(john, eq(phone-num(mary), phone-num(bill)), t_a)

Robin will have discovered (14') while deriving (18) and (15') is just the *belief* version of (15). Assuming that Robin realizes that time t_b is later than t_a by interval_j (but earlier than the current time, t_c), and recalling our previous assumption about "presently-in-mind", Robin may now derive the following (on the basis of (14'), (15'), (25), A14, and our shorthand conventions):

(26) *reasonable-assumption*(*just-inferred*(john, eq3, *set-off*(eq1, eq2), t_b)

Now, because Robin understands the semantics of 'reasonable-assumption', she moves from (26) to (27) (below)

(27) *just-inferred*(john, eq3, *set-off*(eq1, eq2), t_b)

while making a mental note that (27) is likely, but not certain. From (27) and NIM A8, Robin derives *cand*(eq3, john, t_b), which becomes, when the shorthand is dropped:

(28) *cand*(eq(c, phone-num(bill)), john, t_b)

Robin's next inference requires NIM A12, which is,

A12. [*cand*(f, i, t₁) ∧ *later*(t₁, t₂) ∧ *successful*(i, f, t₂)] → *should-be-inserted*(i, f, t₂)

Now, Robin can use A12 by noting that t_c (which is the current time) is a later instant than t_b, and by assuming that John's *candidate* for belief (expressed in (28)) is successful. On the basis of analogy with her own experience, Robin may be assumed to suppose that *others* do not subject trivial conclusions to scrutiny processes, and so Robin assumes that eq(c, phone-num(bill)) is a *successful* candidate for John. Using (28) and A12, Robin may now derive

(29) *should-be-inserted*(john, eq(c, phone-num(bill)), t_c)

but in doing so she propagates to (29) the (small) uncertainty associated with (27) and her additional

assumptions. In order to derive that John actually believes $eq(c, phone\_num(bill))$ at t_c , Robin must assume that John is sufficiently rational to believe what he rationally should believe. Because most preconditions of rational belief have been satisfied by the time an intension is labelled a *candidate*, and the one remaining precondition is satisfied when the candidate is judged *successful*, the predicate *should-be-inserted* (in A12) carries substantial force; it would be irrational for an agent not to adopt as beliefs those intensions which have the property "should-be-inserted". Robin would be justified, therefore, in adopting as a *working assumption*:

A15 $should\_be\_inserted(agent, f, t) \rightarrow B(agent, f, t)$

Given A15 and (29), Robin derives $B(john, eq(c, phone\_num(bill), t_c))$. This in turn leads, via existential generalization, to

(30) $\exists x \{ B(john, eq(x, phone\_num(bill)), t_c) \}$

which is what we want to prove. Note that (30) describes John's belief in an indefinite form (the denotation of x is unknown). Furthermore, since the embedded 'eq' predicate returns an *intension*, Robin's concluding (30) does not commit her to any views about John's representation language, except the view that John's language is rich enough to express *ground instances* of (30). But that richness is already ensured by the initial premises, (14) and (15).

3.2 Extending Inter-Agent Reasoning

In the foregoing derivation, the reader will no doubt have noticed that the *projected* inference

$$(eq(a,b), eq(b,c), \text{therefore } eq(a,c))$$

is so trivial that Robin's projection of that inference to John (via A13) involves very little uncertainty. However, when we move to slightly less trivial inferences, the uncertainty involved markedly increases (at least when human agents are involved). Thus, if the *projected* inference had involved an application of *modus ponens* and *universal instantiation*, the conclusion that a second agent (John) had performed the inference in a given time interval becomes noticeably uncertain. If the projected inference involved a short series of immediate inferences (say three applications of *modus ponens*), the projected conclusion becomes even more uncertain. There may be highly specialized applications (e.g., within distributed computing networks) where each of a pair of agents can safely assume that the other will perform immediate inferences to some specified depth, but such applications are uncommon. It is for this reason, among others, that our discussion has been confined to "trivial-consequences". For, although this concept

is admittedly vague, it does have some clear instances (e.g., the examples so far discussed), and it is doubtful whether *non-trivial* inferences can safely be projected onto other cognitive agents, except in exceptional cases.

Now, some of the foregoing qualifications and uncertainties disappear when our concern is not with what an agent *believes* at a given time, but with what the agent *could justifiably* come to believe.¹⁶ It is not clear how useful it would be, in a *general* way, to equate 'X believes *p*' with 'X could *justifiably* come to believe *p*', but if *that* is our goal, then our previous proof, in which Robin derives (30), could easily be revised to achieve this end. We need only replace all occurrences of 'trivial-consequence' (in our axioms and derivation) with 'can-derive' to obtain the desired result. For example, the intermediate conclusion at (23) would become

can-derive(robin, set-of(eq1,eq2), eq3, interval_j)

and similarly for (25). The revised version of axioms A13 and A14 would still be justified on *inductive* grounds in cases where the agents involved have equal deductive powers. It is interesting to note that, when Robin and John are both endowed with A1-A15, and when both agents have knowledge of the other's *initial* beliefs, John should be able to discover that Robin *could deduce* (30), and hence that (30) is one of Robin's beliefs (in this extended sense of *potential* belief). Robin, in turn, may infer that John could make this latter deduction, and so on, as long as the exercise remains fruitful.

Before leaving the topic of inter-agent reasoning, it should be emphasized that axioms A13-A15 are only relevant to agents who have introspective abilities and "uncertainty propagation" mechanisms. (For this reason, A13-A15 have been presented separately from the NIM axioms, A1-A12.) Moreover, as mentioned, the axioms are justified on inductive grounds, and would not be applicable in possible worlds where agents vary widely in the richness of their representation languages and logics. However, A13-A15 are applicable in a *wider* range of circumstances than the purely syntactic models of Haas and Maida, which presuppose that agents have identical representation languages. Furthermore, agents who use NIM and A13-A15 are forced to make *explicit* assumptions regarding some of the deductive abilities and inferences present in other agents. Thus, the agents themselves are forced to face the contingencies which

¹⁶Indeed, only this latter sense of "could come to believe" seems compatible with the systems presented by Haas (1986) and Maida (1986). For neither Haas nor Maida show any concern for the *complexity* of the argument being projected from one agent to another. While it may be defensible to suppose that agent X, whose abilities are presumed equal to agent Y's, *could* discover any given proof that Y has discovered, it is clearly dangerous to suppose that agent X *will* discover each such proof. One can construct artificial situations in which agents X and Y each have identical tasks and inputs. In such a case the danger just described may be greatly diminished. However, even in these extreme situations, one of the agent's may suffer a "system failure" which the other does not. Such a failure may well lead to an asymmetry in the inference processes of the two agents.

are frequently ignored or left implicit by system builders who presuppose highly artificial domains. Finally, it should be noted that although A13-A15 have (because of their *contingent* status) been presented separately from A1-A12, their addition to the theory $\mathcal{T}$ would not affect the soundness or completeness of $\mathcal{T}$. For A13-A15 are merely additional premises expressed in $\mathcal{L}$, and do not involve the distinguished truth predicate, T .

3.3 Applications of the Truth Predicate

We have previously noted the relationship of *truth* to the concept of knowledge, and have expressed this relation in axiom A3. Moreover, inference rules T1 and T2 capture important aspects of commonsense reasoning about truth. We now briefly consider ways in which T1 and T2 may be useful in acquiring and transmitting inductive beliefs. Suppose our robot Bill hears a number of Mary's remarks (call them $R1$, $R2$, $R3$, etc.) concerning the habits of waterfowl. And suppose that Bill remembers these remarks, and later has an opportunity to verify them. In time, Bill discovers that each of Mary's remarks describes what is the case. That is, Bill observes that $R1$, $R2$, $R3$,.... Using rule T1, Bill infers $T(R1)$ from $R1$, $T(R2)$ from $R2$, etc. As time passes, Bill has a growing list of observations of the form:

$$\text{asserts}(\text{mary}, p) \wedge \text{concerns}(p, \text{waterfowl}) \wedge T(p).$$

Eventually, Bill makes an inductive leap and postulates

$$(31) \quad \forall p \{ [\text{asserts}(\text{mary}, p) \wedge \text{concerns}(p, \text{waterfowl})] \rightarrow T(p) \}$$

The use of rule T1 has been a necessary step in the formulation of Bill's generalization. Now, suppose Bill conveys (31) (in essence) to Jane, who has a growing interest in waterfowl. If Jane believes Bill, she may now regard Mary as an authority on waterfowl. When Jane is told that Mary asserted q about waterfowl, Jane may use (31) to infer $T(q)$, and then use T2 to infer q . In this manner, q may enter Jane's belief base, which in turn permits Jane to act on q . (e.g., Jane now looks for woodducks in the first week of November). Although the preceding example by no means exhausts the ways in which we use the concept of truth, it does suggest that $\mathcal{L}$, T1, and T2 have a useful role to play in the explication of that concept.

4. THE SCOPE OF NIM

The twelve axioms of NIM are intended to specify *what* an agent *should* believe at each moment in its life, given the inferences the agent has made and *accepted* as valid. In presenting the model, we have not assumed that an agent's beliefs are consistent, or that the agent's inferences are usually valid (even by the

standards of "commonsense reasoning"). Because NIM is neutral with regard to the *degree* of logical expertise of an agent, it may be used to model either the belief states of a careless, inconsistent human or those of a reliable, algorithmically driven ATMS. In the former case, a reasonably fine-grained interpretation of 'intension' may be chosen, so that axioms 2 and 3 yield equivalence classes of believed formulae which, intuitively, seem to be equivalent. For example, we may want 'John spent a fortnight in Vancouver' to express the same intension (and hence the same belief) as 'John spent two weeks in Vancouver'. But we would not wish to allow every pair of logically equivalent sentences to have identical intensions and belief status. In fact, when human agents are involved, we cannot even suppose an agent's beliefs to be closed under simple equivalences such as de Morgan's laws. As many logic instructors will attest, a student may well believe $\neg p \wedge \neg q$ without believing, at the same time, $\neg(p \vee q)$. Thus, for modelling human belief systems, we may choose a theory of intensions which would admit the synonymy of the "fortnight" examples while barring the de Morgan example.

By contrast, when we are attempting to model the "beliefs" of *certain* AI systems, e.g., an ATMS, a broader interpretation of intensions is appropriate. To make this clear, we digress briefly to describe the beliefs of an ATMS. At first glance, it may seem odd (over and above the anthropomorphism which permeates our discussion) to attribute beliefs to an ATMS itself, for the ATMS appears to manage the beliefs of a client system. The matter is not so simple, however. For an ATMS has its *own* database of assertions, concerning the dependencies of the client's conclusions upon the client's assumptions (hypotheses), and the assertions in the ATMS database are taken to be *true* by the ATMS. By contrast, the client served by an ATMS need not be committed to any of the hypotheses in the "assumption sets" (or "environments" in de Kleer's terminology) whose ramifications the client explores with the assistance of the ATMS. Thus, it seems more appropriate to describe the assertions in an ATMS database as *beliefs* than to describe the client's hypotheses as beliefs.

In de Kleer's system the assertions in the ATMS database are called nodes, and have the form

(32)

$$\langle \text{datum}, \{\text{assumptions}\}, \{\text{justifications}\} \rangle$$

Here, 'datum' refers to a sentence provided by the client problem solver, '{assumptions}' denotes the set of *alternative sets* of hypotheses upon which *datum* ultimately depends, and '{justifications}' lists *propositional* abstractions of the reasoning by which the client obtained *datum*. If the datum in question is itself a hypothesis, rather than a derived conclusion, then *(assumptions)* and *(justifications)* list the

datum itself as sole assumption and justification. Now, note that (32) could be represented in $\mathcal{L}$ as a simple predication:

(33)

node(depends(datum, {assumptions}, {justifications}), <status>).

which simply asserts the existence of a node associated with the named datum, assumptions, and justifications. In (33), the expressions '*{assumptions}*' and '*{justifications}*' are placeholders for constants (i.e., names supplied by the ATMS) which denote sets that the ATMS manipulates. Also, the term *<status>* is a placeholder for one of the constants 'in', 'out', 'true', 'false'. These constants have a special meaning for the ATMS, but for present purposes we need not consider their precise role. Suffice it to say that they reflect the ATMS stance towards the *client's* datum.

Now, since the database maintained by the ATMS consists of a list of entries similar to (32), we may translate this database into a list of sentences having the form of (33). Thus, we obtain a BB in the language of NIM. The axioms in the BB correspond to nodes created by the ATMS when it is told the basic hypotheses of the client problem solver. Over time, the ATMS will insert and retract sentences like (33) from the BB, depending upon the inferences the ATMS makes. The ATMS makes its inferences partly on the basis of what inferences the client makes.

Now, what concerns us at the moment is that assertions in the ATMS BB resemble second-order beliefs, for though they are not literally about the *beliefs* of the client, they are about propositions which the client entertains. Thus, the first argument of 'depends' in (33), (i.e., 'datum'), denotes a client's hypothesis or derived consequence. The question arises, should 'datum' denote a particular *sentence* in the client's representation language, or should it denote an equivalence class of sentences which all express the same intension? A remark of de Kleer's (1986) will help to answer this question. He notes that "it is crucial that two data which are the 'same' be associated with the same ATMS node. This can only be achieved by some indexing scheme within the problem solver such that all forms of the same datum are associated with the same node. If this is not done, and each new derived datum is automatically associated with a new node, then many of the advantages of the ATMS disappear." Although de Kleer does not explain what he means by "all forms of the same datum", his intended meaning seems to be that distinct formulations of a proposition, which are equivalent from the perspective of *truth maintenance*, should be classed together by the client problem solver. Only by so doing can one ensure that derived *conclusions*, which are "equivalent", are assigned the same *dependency* relationships *vis a vis* the hypotheses sets.

This in turn facilitates the discovery of inconsistencies buried within any hypothesis set managed by the ATMS. Discovering these inconsistencies is the *raison d'être* of the ATMS.

Now, let us assume for the moment that sentences of the client's representation language can, in practice, be partitioned into classes which are "equivalent from the perspective of truth maintainance". Let us also interpret 'intension' so that all sentences in a given equivalence class are said to express the same intension. Then sentences of $\mathcal{L}$, such as (33) above, perspicuously represent what the nodes in the ATMS database *should* express (and, as we later show, the axioms of NIM model the state transitions of the ATMS database). For we need only interpret

(33)

node(depends(datum, {assumptions}, {justifications}), <status>).

as expressing the ATMS belief that a particular assertion of dependency has the status indicated by <status> to obtain the desired result. We must consider, however, whether our assumption concerning equivalence classes is justified. That is, is it true that the sentences of every client's representation language can, in practice, be partitioned into classes which are equivalent from the perspective of truth maintainance? A great deal depends upon what we take this equivalence relation to be. What sentences *should* be considered equivalent from the truth maintainance perspective? Ideally, the required sense of equivalence is full *logical equivalence*, for truth maintainance is concerned with the *consistency* of a set of hypotheses. An ATMS aims at *local* consistency (within an environment) while the standard TMS seeks global consistency within an entire database. In either case, if a given sentence is retracted in the hopes of restoring consistency, success can follow only if all sentences logically equivalent to the retracted sentence are also retracted.¹⁷ We know, however, that for sufficiently rich representation languages, we cannot guarantee that all logically equivalent sentences can even be identified. Therefore, if an ATMS (indeed *any* TMS) is to be effective, we require a sense of "equivalence" for which the client possesses not only a decision procedure, but a decision procedure which operates in feasible time. This does not entail that we exclude from consideration all clients who reason in languages as rich as FOL, for there may yet be useful subclasses of the client's language which can be treated as equivalence classes. Furthermore, the client may well be prepared to accept that the ATMS can only strive for consistency *relative* to the client's specification of equivalence.

¹⁷de Kleer's ATMS is not designed to guarantee the *absolute* consistency of even a local "environment". It can only guarantee consistency relative to the *known* consequences of an environment.

Consider the following example: client Jones reasons in a rich language (such as $\mathcal{L}$) but specifies that a pair of sentences, S_1 and S_2 , will count as *intensionally* equivalent just in case S_1 and S_2 both compile into the same set of *clauses* (via the standard clause transformation algorithm), once all non-primitive predicates (such as *brother(x)*) are replaced by their primitive definitions (e.g., $\text{male}(x) \wedge \text{sibling}(x)$), and once the *literals* in each resulting clause are alphabetically reordered.¹⁸ Such a definition of intensional equivalence would at least ensure that *many* logically equivalent sentences are placed in the same equivalence class. The definition also ensures that a "canonical representation" exists for each sentence in the language. To be sure, such a canonical representation has little plausibility as a sentence of Jones' "language of thought", but *consistency*, rather than psychological plausibility is the goal when dealing with ATMSs. Given the existence of a canonical representation for each sentence in $\mathcal{L}_{\text{Jones}}$, every hypothesis Jones entertains may be "compiled" into its canonical representation, so that all hypotheses which express the same *intension* (in this special sense) may be stored together. This enables the ATMS to ensure a far greater degree of consistency than it otherwise could.

It is interesting to note that even if Jones' representation language is a weak, tractable, propositional logic the strategy of choosing some alphabetically ordered (Boolean) normal form as the canonical representation of intensions is advantageous. For the existence of such canonical forms permits each sentence to be compiled and stored at an appropriate location, at the time the sentence is first encountered. Whether the cost of compiling sentences into their canonical "intensions" is worth the gain in consistency, which the ATMS can consequently deliver, must be decided in particular applications.

5. NIM as a MODEL of an ATMS

In the preceding we have seen that both standard and non-standard interpretations may be assigned to 'intension', where the choice of interpretation reflects the intended application. In the following section we consider how NIM may be used to model the state transitions of an ATMS database.

5.1 NIM and the "Beliefs" of an ATMS

In order to describe the relationship between NIM and the beliefs of an ATMS, it is necessary that we have a somewhat more detailed outline of the operations of an ATMS. In essence, the actions of an ATMS are always a response to an input provided by the client problem solver. Initially, the assumptions of the client are given to the ATMS, and for each such assumption the ATMS creates a node. From

¹⁸Clearly, we are assuming that at least some non-primitive predicates exist. There are some, e.g., Quine, who question the existence of non-primitive predicates in natural language, but even Quine concedes the existence of such predicates in artificial languages.

NIM's viewpoint this means that the ATMS has acquired a set of axioms (beliefs) which are represented in BB by a series of sentences having the following general form:

(34)

node(depends(datum, {environments}, {justifications}), status)

From a logical standpoint, before instances of (34) are inserted in the BB, they have the NIM status of "just-presented", and by axiom A9, these sentences are merely candidates for entry into the BB. But, since the corresponding ATMS nodes merely describe the client's *initial* assumptions, the ATMS does not subject these nodes to a "questioning" phase, because the ATMS has not yet discovered any inconsistent environments (assumption sets). Thus, each of these nodes is regarded by the ATMS as successful, and so, by NIM axiom A12, the NIM counterparts of each of these nodes (which are instances of (34)), should each be inserted in the BB. Since the ATMS is constructed to be rational, it adopts the beliefs it should adopt. We will assume therefore, that when we can derive via A12 a conclusion such as

(35)

should-be-inserted(atms, node(.....), t)

that the indicated node is added to the ATMS' BB. (hereafter, we shall refer to instances of (34), above, simply as "nodes").

Now, once the ATMS has adopted a set of beliefs concerning the status of its client's assumptions, its future beliefs will be formed in response to input from the client (such *input* asserts that the client has derived a conclusion *c* from some subset of the client's assumptions.) The ATMS will process this client information and create an ATMS node describing the dependency of *c* upon the client's assumptions and justifications. At this point the ATMS status of this node is unknown. From NIM's standpoint, the ATMS has acquired a belief, via axioms A9 and A12 (as before), where the belief is represented in BB by something like:

(36)

node(depends(c, {environments}, {justifications}), unknown)

A belief such as (36) triggers an inference process in the ATMS, wherein the set of environments is examined to determine whether any environment supporting *c* is inconsistent. The outcome of this inference is always that some new node is inferred whose ATMS status is known, and whose environments are not known by the ATMS to be inconsistent. When this happens, ATMS' belief in (36)

is simply retracted, (36) disappears from the BB, and there will be a new node that ATMS has just inferred. By A8 this new node becomes a candidate. Suppose this candidate has the form:

(37)

$$\text{node}(\text{depends}(c, \{E_1\}, \{J_1\}), \text{in})$$

Because the ATMS is designed so that inferences from beliefs like (36) to beliefs like (37) are correct and not questioned, the candidate (37) is successful and, by NIM axiom A12, (37) enters the BB. (Of course, the ATMS' own database does not contain expressions like (36) and (37); we are describing NIM's formal equivalent of the *processes* and contents of the ATMS database.)

Now, while the ATMS infers (37) from (36), it may discover that some environment mentioned in (36) is inconsistent. For each such inconsistent environment, E_i , the ATMS acquires a new belief, via its *own* inferences, which is expressible in $\mathcal{L}$ as

(38)

$$\text{inconsistent}(E_i)$$

Since the ATMS infers sentences like (38), and is not designed to question these inferences, conclusions like (38) are "successful" (from NIM's standpoint), and enter BB via NIM axioms A8 and A12. Once the ATMS has discovered some inconsistent environment, E_i , it initiates a search process for those nodes (in its own database) which assert a dependency relationship between a client's datum and E_i . All such nodes are modified by having E_i removed from their *environment* lists. From NIM's standpoint, this means that the ATMS has engaged in a series of inferences which lead to a series of retractions in BB, and the acquisition of new beliefs having the general form: $\text{node}(\dots)$. These new beliefs enter the BB via A8 and A12, just as (38) did above. For example, if at time t_1 the ATMS is searching for nodes containing the inconsistent environment, E_3 , and it discovers such a node (N) at time t_2 , then soon after t_2 the ATMS acquires a *meta-belief* which is represented in $\mathcal{L}$ by a sentence of the form:

(39)

$$B(\text{atms}, N, t_2)$$

A metabelief such as (39) enters the BB via axioms A9 and A12. On the basis of (39) and " $\text{inconsistent}(E_3)$ " the ATMS infers that node N must be retracted and some new node, N' , is inferred. Since the new node, N' , was inferred by a process which ATMS does not question, the NIM counterpart of N' straightforwardly enters the BB by axioms A8 and A12.

By this point, it has hopefully emerged that all "beliefs" of an ATMS may be modelled by NIM. The contents of the ATMS database at any given time correspond to assertions in a BB_{NIM} . Since all ATMS operations can be assimilated to the *formal* paradigm of *inferring* what nodes to insert and retract, and inferring what sets of assumptions are inconsistent, it follows that the state transitions of an ATMS database can be modelled by NIM.

Before we leave the topic of ATMSs, it is worth noting that the ATMS can be used as a standard TMS in which the consistency and consequences of a *single* set of assumptions are explored. Indeed, the ATMS does not bother to check whether its client is exploring the ramifications of a *single* set of assumptions or of multiple sets. It follows then, that NIM can be used to model the belief transitions of at least one type of TMS, and there seems to be no reason why it should not be used to model other standard TMSs, such as (Doyle, 1979).

6. CONCLUSION

A generalized first-order model of belief and knowledge has been presented which simultaneously avoids logical omniscience and the excessively fine-grained "syntactic approach". The model, NIM, is intensionally based, and sanctions inferences involving nested epistemic attitudes, with different agents and different times. Unlike the earlier systems of Perlis (1985) and Haas (1986), NIM attains its rich expressive power without resort to encumbering quotation marks and concatenation functions. Moreover, because NIM is grounded upon intensions, it provides a framework in which agents may reason about the beliefs of another agent while remaining neutral with respect to the syntactic forms used to express the latter agent's beliefs.

In section 3 we demonstrated that the axioms of NIM may be used to *justify* conclusions which an artificial agent derives concerning the beliefs of another agent. Because NIM makes explicit many (if not all) of the empirical preconditions of belief acquisition, NIM underscores the limitations of inter-agent reasoning. In normal circumstances (which exclude highly artificial domains), agents cannot form reasonable conclusions concerning the newly *inferred* beliefs of another agent, *unless* only trivial conclusions (of the target agent) are considered, or unless 'belief' is taken to include *potential* beliefs. A robot who employs NIM is made aware of those conditions under which inter-agent reasoning is feasible. Moreover, NIM provides agents with a conceptual map, interrelating the concepts of knowledge, belief, truth, and a number of cognate concepts, such as 'infers', 'retracts', and 'questions'.

From a logician's standpoint, the language of NIM provides a useful tool for doing intensional logic. In addition, the inclusion of a distinguished truth predicate, and the extension of FOL to include the inference rules T1 and T2, provides $\mathcal{L}_{\text{NIM}}$ with an expressive power approaching that of natural language. We have seen that knowledge of how to reason about *truth* permits agents to acquire and transmit useful inductive knowledge.

In section 4 we argued that the *scope* of NIM varies according to the degree of granularity chosen for the class of 'identical intensions'. We have seen that when an intuitive, human-like concept of belief is desired, one should individuate intensions so that (roughly) only those formulae which we intuitively regard as *synonymous* are permitted to have the same intension, and thus express the same belief. We have suggested several semantic theories which yield (approximately) the required criterion of synonymy. By contrast, when NIM is applied to artificial domains, such as 'truth maintainance', a coarse-grained criterion of 'intensional identity' is appropriate. In section 5, we demonstrated that when a liberal interpretation of 'intensional identity' is adopted, NIM serves as a formal model of the epistemic states of an ATMS. Thus, NIM can serve both as a specialized model of the epistemic processes of a powerful AI tool, and as a *general* model of epistemic state transitions. The flexibility of NIM emphasizes our overriding concern with the development of cognitive theories which integrate the goals of AI and Cognitive Science.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the support of the Natural Sciences and Engineering Council of Canada (Grant No. A0899) during the period of this research.

I wish to thank Raymond Jennings, Charles Morgan, and Susan Wendell for their helpful comments on an earlier version of this paper. Also, my sincere thanks to the following for discussions which stimulated many of the thoughts contained herein: James Delgrande, Raymond Jennings, Charles Morgan, Brian Smith, Lenhart Schubert and Susan Wendell.

REFERENCES

- Aleliunis, R., (1988) "A summary of a new normative theory of probabilistic logic", Proceedings of the 4th Workshop on Uncertainty in AI, University of Minnesota, pp. 8-14.
- Barnden, J.A., (1986) "Imputations and explications: representational problems in treatments of propositional attitudes", *Cognitive Science*, Vol. 10, pp. 319-364.
- Brown, A.L., Gaucas, D.E., and Benanav, D., (1987) "An algebraic foundation for truth maintainance", Proceedings of the Tenth International Joint Conference for Artificial Intelligence, Milan, pp. 973-980.
- Cresswell, M.J., (1985) *Structured meanings: the semantics of propositional attitudes*, MIT Press, Cambridge, MA.
- de Kleer, J. (1986) "An assumption-based TMS", *Artificial Intelligence*, Vol. 28, pp. 127-162.
- Doyle, J. (1979) "A truth maintenance system", *Artificial Intelligence*, Vol. 12, pp. 231-272.
- Fagin, R., and Halpern, J., (1985) "Belief, awareness, and limited reasoning", Proceedings of IJCAI, Los Angeles, pp. 491-501.
- Frege, G. (1952) "On sense and reference", in *Translations from the Philosophical Writings of Gottlob Frege*, Oxford.
- Gawron, J.M. (1986) "Types, contents, and semantic objects", *Linguistics and Philosophy*, Vol. 9, No. 4, pp. 427-476.
- Goodwin, J.W. (1985) "A process theory of non-monotonic inference", Proceedings of the Ninth International Joint Conference on Artificial Intelligence, August, Los Angeles, pp. 185-187.
- Haas, A., (1985) "Possible events, actual events, and robots". *Computational Intelligence*, Vol. 1, No. 2, pp. 59-70.
- Haas, A., (1986) "A syntactic theory of belief and action", *Artificial Intelligence*, Vol. 28, 245-292.
- Hadley, R.F., (1986) "Fagin and Halpern on logical omniscience, a critique with an alternative", Proceedings of the Sixth Canadian Conference on Artificial Intelligence, Montreal, May, pp. 49-56.
- Hadley, R.F. (1988) "Logical omniscience, semantics and models of belief", *Computational Intelligence*, Vol. 4, pp. 17-30.
- Hadley, R.F., (1988b) "A default-oriented theory of procedural semantics", *Cognitive Science*, Vol. 13, pp. 107-137.
- Kleene, S.C., (1952) *Introduction to meta-mathematics*, Wolters-Noordhoff Publishing and North-Holland Publishing, New York.
- Konolige, K., (1985) "A computational theory of belief introspection", Proceedings of IJCAI, Los Angeles, pp. 503-508.

- Kripke, S., (1975) "Outline of a theory of truth", *Journal of Philosophy*, Vol. 72, pp. 690-716.
- Lakemeyer, G., (1987) "Tractable meta-reasoning in propositional logics of belief", *Proceedings of IJCAI*, Milan, pp. 401-408.
- Levesque, H.J., (1984) "A logic of implicit and explicit belief", *Proceedings of the National Conference on Artificial Intelligence*, Austin, Texas, pp. 198-202.
- Lewis, D. (1976) "General semantics", in *Montague Grammar*, (ed.) Partee, B., Academic Press, New York.
- Maida, A.S. (1986) "Introspection and reasoning about the beliefs of other agents", *Proceedings of the Cognitive Science Society*, pp. 187-195.
- Montague, R., (1970) "Universal grammar", *Theoria* 36.
- Moore, G.E., (1942) "Reply to my critics", In *The philosophy of G.E. Moore*, (Ed.) P. Schilpp, Northwestern University Press, Evanston and Chicago.
- Pearl, J., (1986) "Fusion, propagation, and structuring in belief networks", *Artificial Intelligence*, Vol. 29, pp. 241-288.
- Perlis, D., (1985) "Languages with self-reference I: foundations", *Artificial Intelligence*, Vol. 25, pp. 301-322.
- Tarski, A. (1936) "Der wahrheitsbegriff in den formalisierten sprachen", *Studia Philos.*, Vol. 1, pp. 261-405.
- Wilks, Y. & Ballim, A. (1987) "Multiple agents and the heuristic ascriptions of belief", *Proceedings of IJCAI*, Milan, pp. 118-124.

Appendix A

Proof of the completeness of $\mathcal{T}$.

The language of $\mathcal{T}$ is simply the language of PC. No new vocabulary has been introduced, although the interpretation of $\mathcal{T}$ is fixed. Moreover, the interpretation of $\mathcal{T}$ is entirely consistent with the principles of Tarskian interpretation. Now, the inference rules of FOL are a subset of $\mathcal{T}$, and these rules are known to be complete for any first-order theory. Thus, these rules ensure that every *universally valid* consequence of $\mathcal{T}$ is derivable in $\mathcal{T}$, since $\mathcal{T}$ is just a first-order theory with additional rules of inference. The two additional rules do not block the application of any of the standard FOL inference rules. However, since we are restricting the models of $\mathcal{T}$ to those where $\mathcal{T}$ has a special interpretation, we must consider whether any *additional* consequences hold in these *restricted* models (hereafter, T-models) which do not hold in the unrestricted set of all models for $\mathcal{T}$. Clearly, the answer is yes. In particular, we know that any T-model which satisfies a sentence of the form $\mathcal{T}(\phi)$ will also satisfy ϕ , and any T-model which satisfies ϕ will satisfy $\mathcal{T}(\phi)$.

Now, to establish the completeness of $\mathcal{T}$ we need to show that if Γ is a set of sentences expressed in $\mathcal{T}$, and ϕ is a consequence of Γ , then ϕ is derivable from Γ . "Consequence of Γ " here includes all sentences

whose truth is *semantically entailed* by the joint assumption of Γ with the special interpretation of T as a truth predicate. In general outline, our proof follows Mate's [1965] version of the standard Henkin completeness proof for FOL. However, modifications are required to the standard proof in order to accommodate our fixed interpretation of T . These modifications will be explained in detail. In the following it is important to bear in mind that the language of T , which is $\mathcal{L}$, contains only the syntax and vocabulary of traditional FOL with functions. The proof of completeness requires the following major lemma, from which completeness follows by a standard, straightforward argument.

Lemma I. If a set of sentences Γ (of $\mathcal{L}$) is d-consistent, then Γ is consistent.

Customarily, Lemma I is proved by means of a special case of I, namely:

Lemma I' If Γ is a d-consistent set of sentences, and every individual constant occurring in a sentence of Γ has an *even* index (e.g., c_2, c_4 , etc.), then Γ is consistent.

Once I' is established, Lemma I follows immediately, since those Γ which satisfy the antecedent of I' are merely notational variants of those which satisfy the antecedent of I. Now, I' can be established as an immediate consequence of the following two lemmas:

Lemma II. If Γ is a set of sentences satisfying the antecedent of I', then there is a maximal d-consistent, ω -complete set of sentences, Δ , such that $\Gamma \subseteq \Delta$.

Lemma III. If Δ is maximal d-consistent and ω -complete, Δ is consistent.

The proof of II is constructive, and requires the existence of an infinite set of individual constants not mentioned in Γ . However, the antecedent of I' was designed to ensure that all constants having *odd* indices do not occur in Γ and are available for use in the construction. The traditional proof of II involves the construction of a maximal d-consistent, ω -complete set Δ which contains Γ . This construction requires that the sentences of $\mathcal{L}$ can be enumerated in a particular sequence, and that the usual rules of FOL be available. However, nothing in the theory T affects this construction, since the language of T is identical to FOL and neither of the rules T1 or T2 interferes with the application of the standard rules of FOL. The reader may readily verify that the standard proof of II is unaffected by our fixed interpretation of the predicate T . For this reason, the proof of II is omitted.

Proof of III.

Let Δ be an ω -complete, maximal d-consistent set of sentences of $\mathcal{L}$. We specify an interpretation I , for $\mathcal{L}$, as follows. The domain of $\mathcal{L}$ will be the set of all sentences, and intensions of sentences of $\mathcal{L}$. Any given individual constant, α , of $\mathcal{L}$, will denote a unique tautology, $P(\alpha) = P(\alpha)$, provided Δ contains the sentence $T(\alpha)$; otherwise, α denotes the necessarily false sentence, $P(\alpha) = \neg P(\alpha)$. Each grounded functional expression, β , which (when removed from its "embedded" context) would *not* constitute a well-formed sentence of $\mathcal{L}$, will denote either a unique tautology involving β , or a necessarily false

sentence involving β , depending upon whether $T(\beta)$ is in Δ (as in the case of α , above). Each embedded functional expression which *could* "stand alone" as a sentence of $\mathcal{L}$ has as denotation its *own intension*. To each n -ary predicate π of $\mathcal{L}$ (not identical to the unary predicate T), I assigns just those n -tuples of objects in the domain of $\mathcal{L}$, $\langle \delta_1, \dots, \delta_n \rangle$, such that there exists an *atomic* sentence in Δ having the form $\pi(\gamma_1 \dots \gamma_n)$, where each term in the n -tuple of expressions $(\gamma_1 \dots \gamma_n)$ denotes a corresponding object in the n -tuple $\langle \delta_1, \dots, \delta_n \rangle$. To the *unary* predicate T , I assigns (recursively, as in section 2.3) the set of those sentences of $\mathcal{L}$ which are true under I , together with the intensions expressed by those sentences.

We now define the *index* of expressions of $\mathcal{L}$ as follows: The index of any sentence or *functional expression* of $\mathcal{L}$ shall be the total number of quantifiers, connectives, and T s occurring in that sentence, including those occurrences embedded in its (functional) arguments. Thus, the atomic sentence $T(\exists x (T(x)))$ has an index of three, $\exists x (P(x))$ has index one and $P(c)$ has index zero.

Now, let the sentences of $\mathcal{L}$ be divided into two disjoint sets. Set S_N contains all atomic sentences *except* those having the form $T(\psi)$, where the index of ψ is greater than zero. Thus, $T(b)$ is in S_N , but $T(\exists x (P(x)))$ is not. Set S_T contains *all other* sentences of $\mathcal{L}$.

We must now prove that

(C)

Every sentence ϕ of $\mathcal{L}$ is true under I (I -true) iff $\phi \in \Delta$.

We begin by proving a special case of (C), namely

(C')

Every sentence ϕ of S_N is true under I iff $\phi \in \Delta$.

Now, clearly, by the construction of I , all true (atomic) sentences in S_N are in Δ , and those atomic sentences in Δ which are also in S_N are true. This is because I was constructed so that every predicate π in $\mathcal{L}$, not equal to unary T , has in its denotation just those n -tuples of objects which are denoted by the n -tuples of expressions which occur as arguments to the π predicates in Δ . Moreover, those atomic sentences in S_N having the form $T(\alpha)$ satisfy (C'), because the denotations of all zero-index arguments to T were chosen so as to ensure this.

We have now established (C'). To establish (C) it remains to show that all sentences of S_T satisfy (C). Now, since every sentence in S_T is either non-atomic or contains arguments of index greater than zero, it

follows that the index of every sentence in S_T is > 0 . Let us suppose that (C) fails to hold for some sentence in S_T . Let ϕ be such a sentence in S_T having lowest index, N . Then ϕ must have one of the eight forms listed below.

Case (1) $\phi = T(\psi)$, where the index of $\psi < N$, but > 0 . (If the index of ψ were zero, ϕ would be in S_N).

Now, by the definition of the extension of T we have that

ϕ is I -true iff ψ is I -true (because ψ is I -true iff the intension of ψ returns 'true' under I , which happens just when ψ is in the extension of T .)

iff $\psi \in \Delta$ (since the index of $\psi < N$, and the induction hypothesis is assumed.)

But, ψ can be in Δ iff ϕ is also, because Δ is d -consistent and ψ and ϕ are each derivable from the other, by T1 and T2. It follows that if ϕ is a sentence of $\mathcal{L}$ which fails to satisfy (C), then ϕ cannot have the form $T(\psi)$, and case (1) is therefore excluded. Note that we have now established that every atomic sentence of $\mathcal{L}$ satisfies (C), for we have proven (C) both in the earlier case where $\phi \in S_N$, and in the present case where $\phi \in S_T$ and ϕ is atomic. No other atomic sentences remain.

Cases (2-6) have the forms listed below:

- (2) $\phi = \neg(\psi)$
- (3) $\phi = \psi \vee \chi$
- (4) $\phi = \psi \wedge \chi$
- (5) $\phi = \psi \rightarrow \chi$
- (6) $\phi = \psi \equiv \chi$

The proofs for the above five cases follow the usual Henkin method. Since these cases do not differ from those found in standard completeness proofs for FOL, the proofs for these cases are omitted.

(7) $\phi = \forall \alpha (\psi)$

(8) $\phi = \exists \alpha (\psi)$

In cases (7) and (8) we must depart from the standard induction process for the following reason. Although the index of ψ is $< N$, not all ground instances of ψ have index $< N$. This is because occurrences of α in ψ may be replaced by a functional expression having an index > 0 . For this reason we now adopt a slightly different strategy. We shall prove, independently of the (previously defined) indexing scheme, that all instances of (7) and (8) satisfy (C), and therefore that the supposed counterexample could not belong to either of these cases.

To begin, recall that we have already established that every atomic sentence of $\mathcal{L}$ satisfies (C). This fact enables us to introduce a notion of *standard* index, defined as follows: all atomic sentences have a standard index of zero. The standard index of any other sentence is simply the total number of quantifiers

and logical connectives which occur in that sentence at levels *above the atomic level*. Using this concept of standard index we establish that *regardless* of the non-standard (previously defined) index of ϕ , ϕ must satisfy (C) if ϕ belongs to case (7) or (8). For, suppose that ϕ does not satisfy (C). Then there must exist some sentence χ (having form (7) or (8)) which fails to satisfy (C), and which has the lowest *standard* index of any such sentence. Call this index M . Now, if χ belongs to case (7), then we have $\chi = \forall \alpha(\Sigma)$, where the standard index of Σ is $M-1$.

Note that, since all atomic sentences have a standard index of zero, all instantiations of Σ (i.e., instances obtained by instantiating on just the α occurrences in Σ) also have index less than M . Hence, by hypothesis, all instances of Σ satisfy (C), i.e., each such instance is in Δ iff that instance is *I*-true. Now suppose that χ is true. Then all of Σ 's instances are true, and so are in Δ . But, since Δ is ω -complete, χ must be in Δ .

On the other hand, suppose χ is in Δ . By ω -completeness, all instances of Σ are in Δ . Since their index is less than M , these instances must all be true. Hence, χ is also true (by definition of $\forall$). Therefore, we have that $\chi \in \Delta$ iff χ is true, contrary to the assumption that χ violates (C). It follows that every sentence of $\mathcal{L}$ having the form of (7) satisfies (C). The argument for case (8) is precisely analogous to that for (7). We conclude, therefore, that the supposed sentence of $\mathcal{L}$ which fails to satisfy (C) cannot fall into case (7) or (8). Since no other cases remain, the supposed counterexample does not exist, and (C) holds for all sentences of $\mathcal{L}$. Moreover, since (C) assumed only a specific interpretation for the expressions of $\mathcal{L}$, and that Δ is *d*-consistent and ω -complete, we have shown that every such Δ is satisfiable. This completes the proof of III.

Centre for Systems Science

- Simon Fraser University established the **Centre for Systems Science** as a provincial centre for advanced study. The objective of CSS is twofold: coordinate and combine advanced systems research at SFU, and encourage application of research results within industry in order to further the technological development of British Columbia.
- CSS has assembled three research groups to streamline its efforts. The orientation of these groups is as follows: intelligent systems, microelectronics, and computer and communications systems. Because of shared research interests, this latter group merges with LCCR (**Laboratory for Computer and Communications Research**) for the publication of technical reports.
- CSS's three groups involve over 80 skilled faculty researchers principally from SFU's Schools of Engineering Science and Computing Science, but also draw substantially on Mathematics, Physics, Geography, Philosophy, Linguistics, Psychology, Communication, and Kinesiology.